16th International Conference on Autonomic and Autonomous Systems
ICAS 2020

September 27, 2020 to October 01, 2020 - Lisbon, Portugal
and online due to COVID-19

Keynote:
Achieving Trustworthy Autonomous Systems through Autonomic and Apoptotic Computing

Roy Sterritt

r.sterritt@ulster.ac.uk

@RoysterUlster

Ulster University

"Short" Bio

Roy Sterritt,
Ulster University, Northern Ireland.
r.sterritt@ulster.ac.uk @RoysterUlster

- Biography—Roy Sterritt is a member of Faculty in Computing and Engineering at Ulster University. He spent several years in industry with IBM, first at their UK headquarters in Portsmouth, and then at the IBM Hursley Labs in Winchester. Initially he was a Software Developer in their KBS department but then became a Product Development Manager with responsibility for tools to support risk assessment and project management in personal and mobile environments which were used widely in the UK and US. Roy's academic research career began in 1996 when he was appointed to the first of a series of joint University of Ulster and Nortel research projects investigating parallel, automated and intelligent approaches to the development and testing of fault management telecommunications systems.

- Roy's main focus of research is Systems and Software Engineering of Autonomic (Self-Managing Computer-Based) Systems, essentially a research area developed from a call from industry to deal with the complexity and total cost of ownership of our systems of systems (IBM 2001). To date he has 200+ publications in the field including research collaborations with NASA, IBM TJ Watson Center, BT, SAP, HP and Core Systems as well as many academic partners. The research with NASA also lead to 16 US patents. He was the founding chair of the IEEE Task Force and subsequently Technical Committee on Autonomous & Autonomic Systems and elected chair of IEEE Technical Committee on Engineering of Computer-Based Systems. He has held many other IEEE roles such as; IEEE CS Publications board member, chair of the Conference Publications Operations Committee (CPOC); served on the IEEE CS Technical & Conferences Activities Board (T&C Excom and Opcom) and chaired the Conference Advisory Committee (CAC). He has been appointed to the many editorial boards including the NASA Journal on Innovations in Systems and Software Engineering, ACM Transactions on Autonomous and Adaptive Systems (TAAS), AIAA Journal of Aerospace Computing, Information, and Communication, Journal of Autonomic and Trusted Computing, and Multiagent and Grid Systems - An International Journal; and served on steering and/or program committees of the majority of the conferences in his field at some stage during the last 20 years.

- This extensive research community activity and NASA collaboration during the *noughties* lead to the common query from colleagues and management "are you ever at home?". As such Roy scaled back his international activity during the *tens* and took on institutional roles such as; Placement Coordinator-looking after 400+ students while seeking and on year-long industrial placement; Manager of CPPD (Continuous Personal and Professional Development) including developing outreach courses and summer schools in both Computer Science and Space Science. Yet with 16 patents with NASA, Roy also took the opportunity to explore spinning out the Autonomic Research as well as continuing that research, in particular through his PhD students.

2001 RECAP

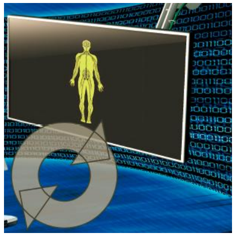
Computer Based Systems are Not Self-Managing

IT Control Room

Best Case

Worst Case

Kaiser et al 2003



2001 RECAP

Autonomic Computing – *Biological Inspiration*

Autonomic Computing, launched by IBM in 2001, has emerged as a valuable approach to the design of effective computing systems.

The autonomic concept is inspired by the human body's autonomic nervous system.

It is the part of the nervous system that controls the vegetative functions of the body such as circulation of the blood, intestinal activity and secretion and the production of chemical ‘messengers’, hormones, that circulate in the blood

Fight or Flight

sympathetic (SyNS)

Rest and Digest

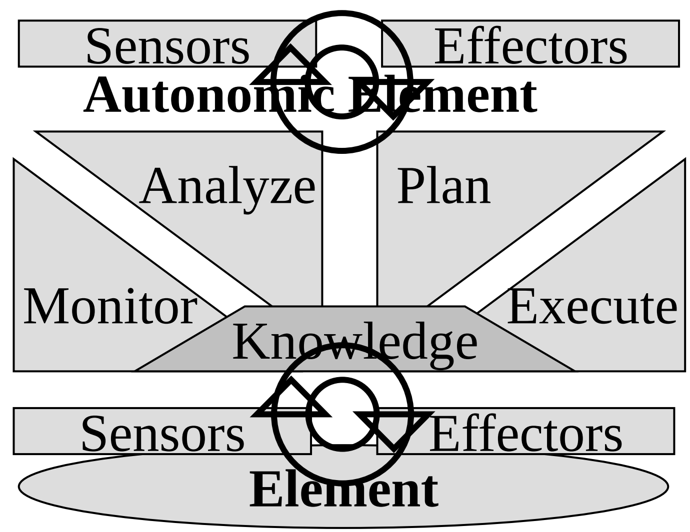
parasympathetic (PaNS)

Autonomic Computing – *IBM Vision*

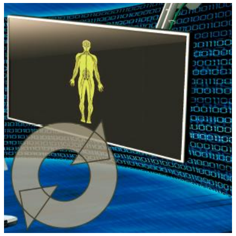
2001 RECAP

- Possess system identity
  - detailed knowledge of components
- Self configure & re-configure
  - adaptive algorithms
- Optimise operations
  - adaptive algorithms
- Recover
  - no impact on data or delay on processing
- Self protection
- Aware of environment and adapt
- Function in a heterogeneous world
- Hide complexity

IBM – AC Manifesto Oct. 2001



IBM – AC Blueprint Apr. 2003



2000's RECAP…

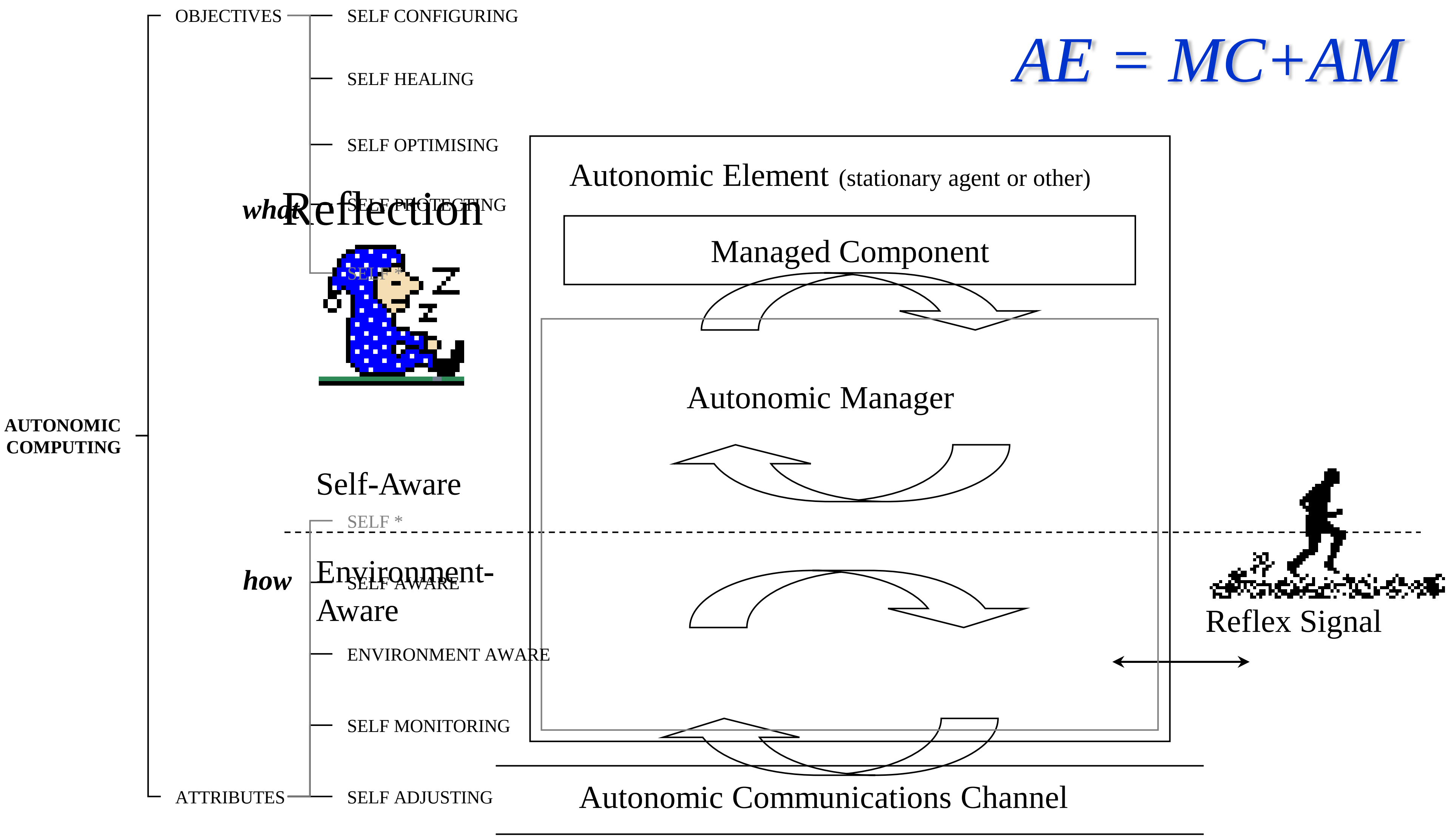
AUTONOMIC COMPUTING

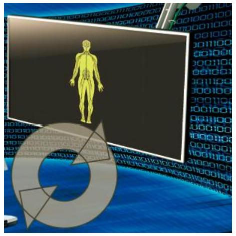
- GOAL
  - SELF MANAGEMENT
- OBJECTIVES — ***what***
  - SELF CONFIGURING
  - SELF HEALING
  - SELF OPTIMISING
  - SELF PROTECTING
- ATTRIBUTES — ***how***
  - SELF AWARE
  - ENVIRONMENT AWARE
  - SELF MONITORING
  - SELF ADJUSTING
- APPROACHES
  - MAKE (SYSTEMS/ SOFTWARE ENG)
  - ACHIEVE (ADAPTIVE LEARNING)

The AC Tree

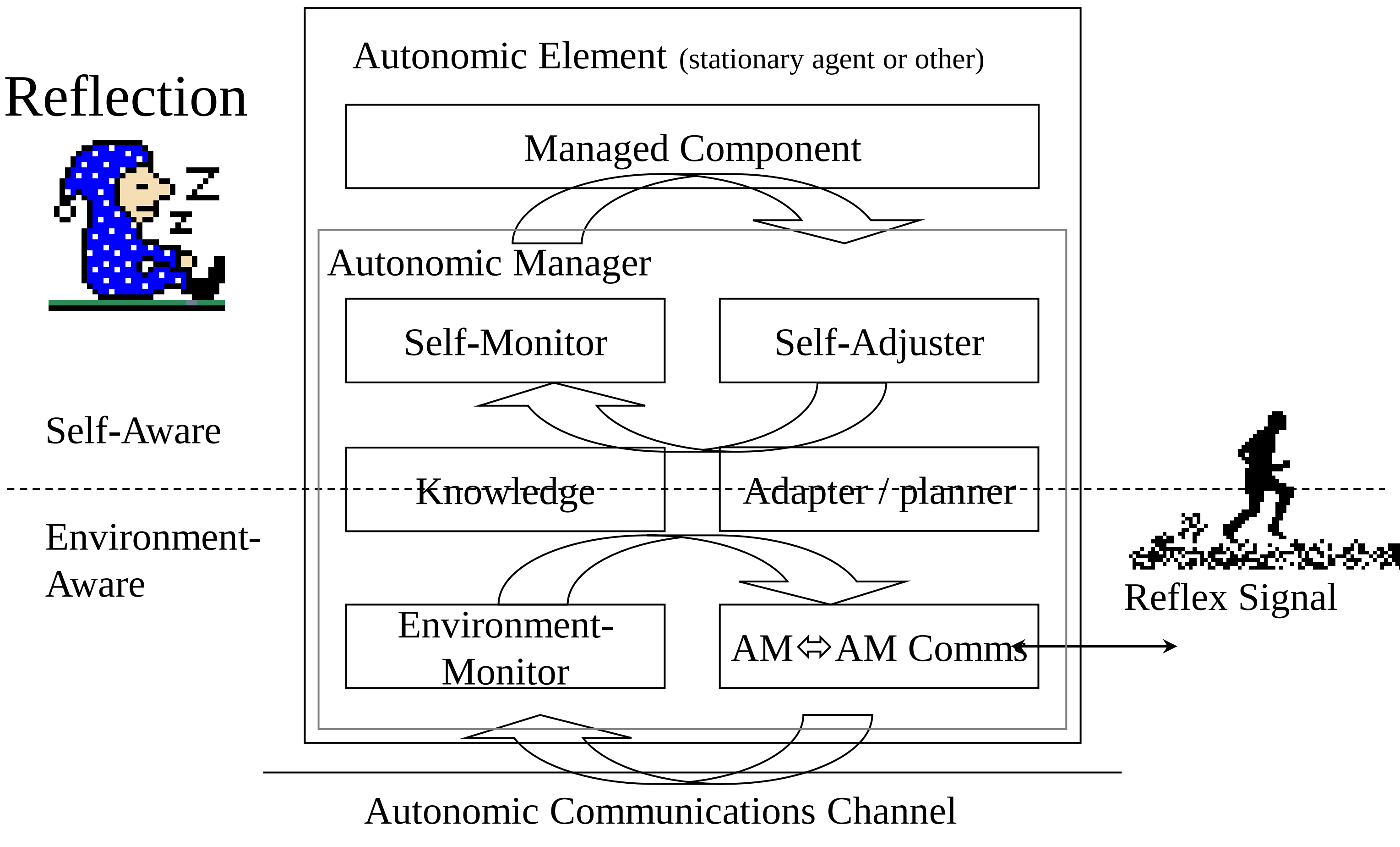
Sterritt et al ECBS 2003

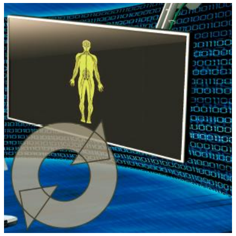
Autonomic Computing – *The Autonomic Element*



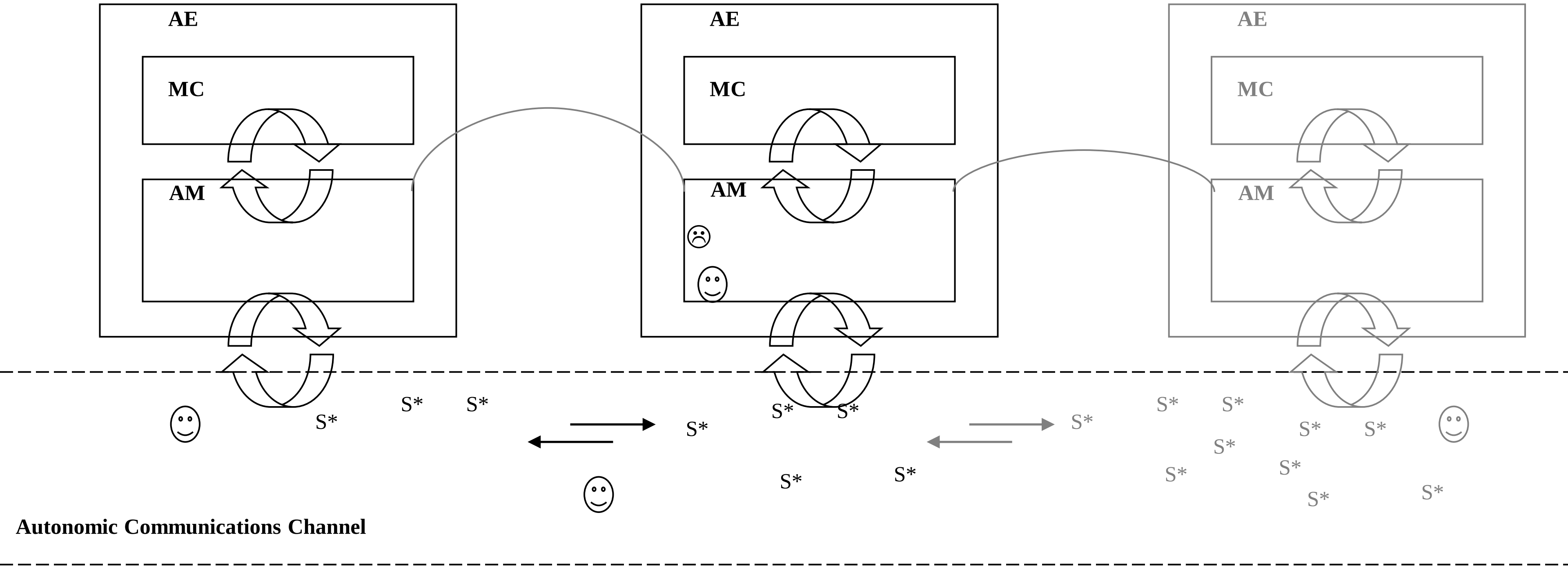


Autonomic Computing – *The Autonomic Manager*



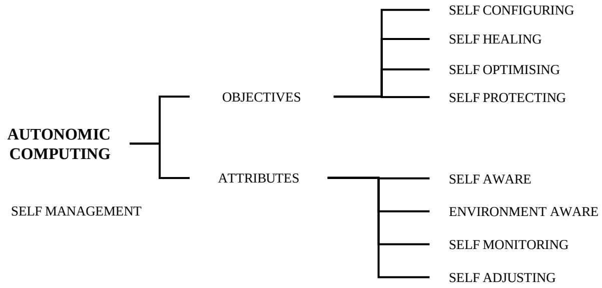


Autonomic Computing – *The Autonomic Environment*

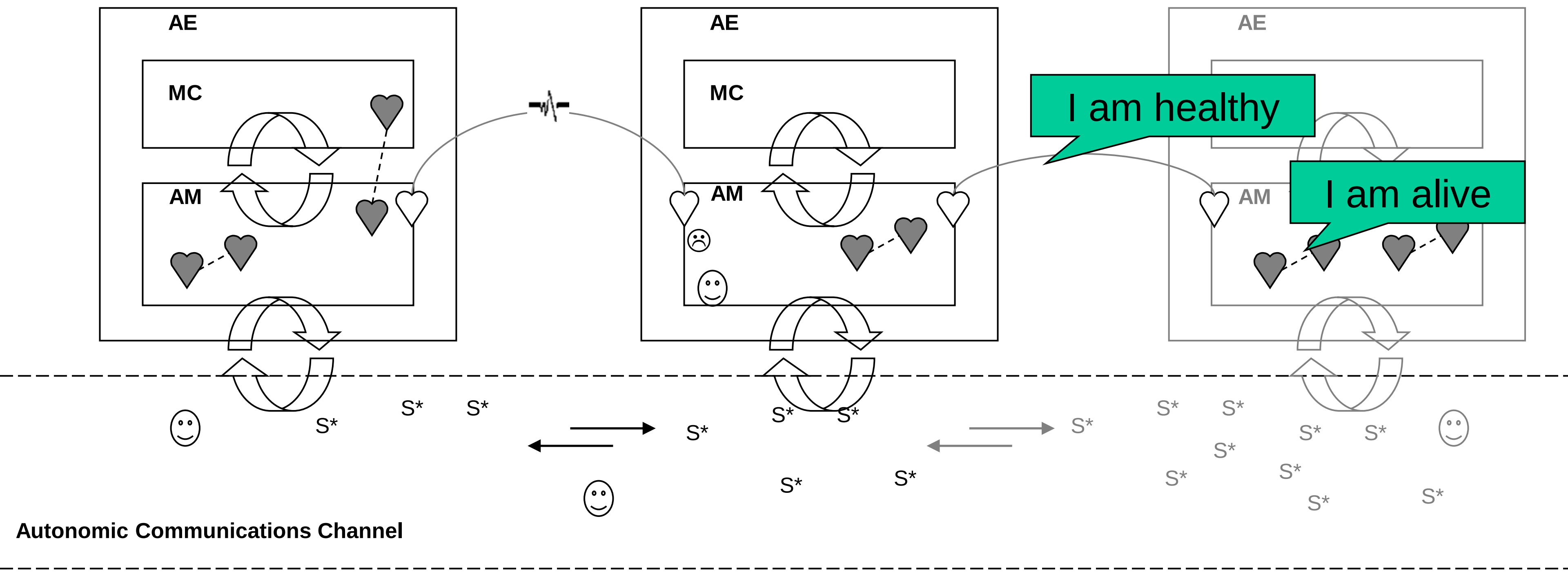


Key

S*	**Self-* event messages**	**AE**	**Autonomic Element (AM+MC)**
		MC	**Managed Component**
		AM	**Autonomic Manager (Stationary agent)**
☺	**Autonomic Agent (Mobile agent)**		



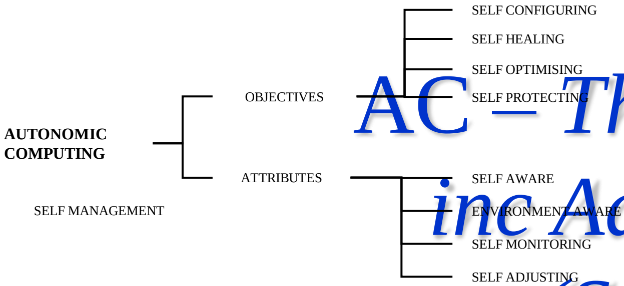
Autonomic Computing –
The Autonomic Environment

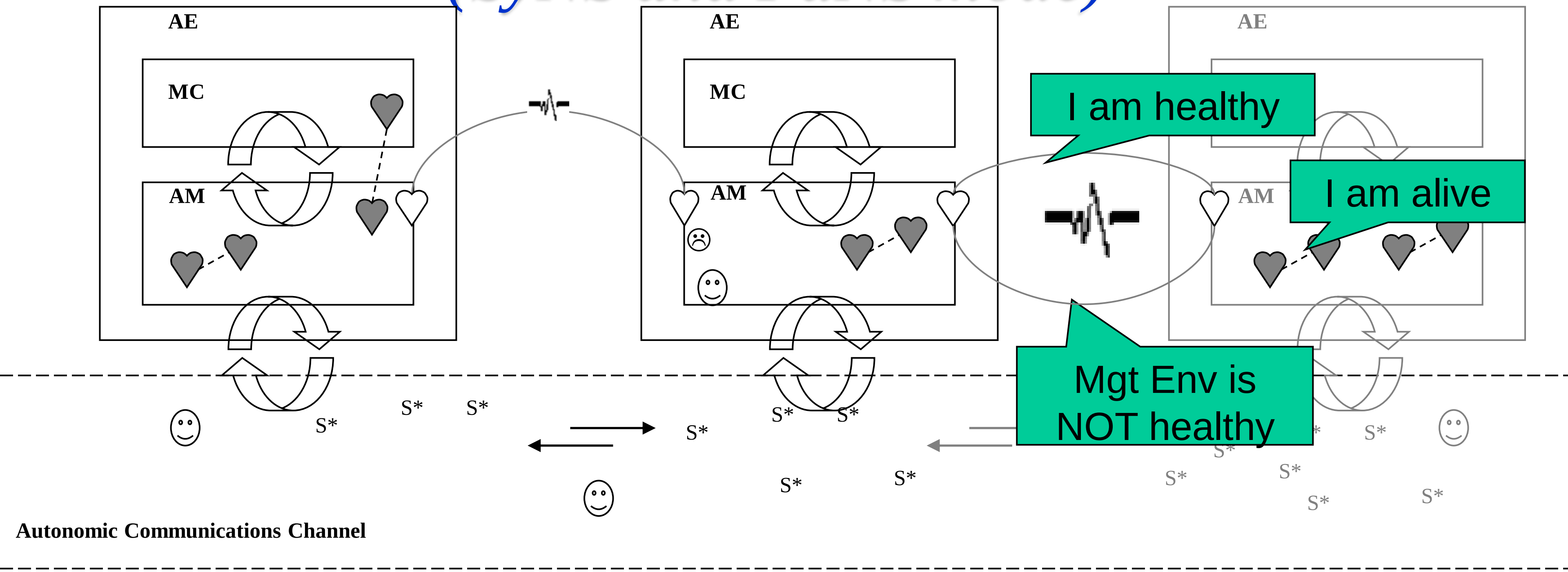


Key

S*	**Self-* event messages**	**AE**	**Autonomic Element (AM+MC)**
♡	**Pulse Monitor (PBM)**	**MC**	**Managed Component**
♥	**Heart-Beat Monitor (HBM)**	**AM**	**Autonomic Manager (Stationary agent)**
☺	**Autonomic Agent (Mobile agent)**		

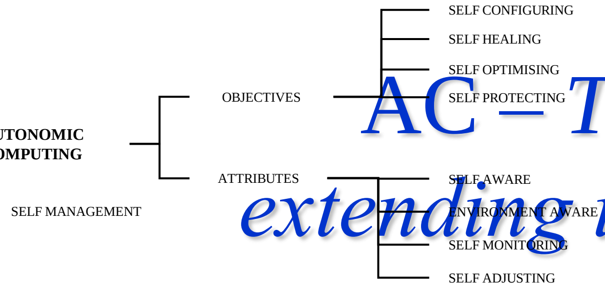
AC – The Autonomic Environment inc Adaptive pulse monitoring (SyNS and PaNS mode)





Key

S*	Self-* event messages	AE	Autonomic Element (AM+MC)
♡ -√-	Pulse Monitor (PBM)	MC	Managed Component
♥	Heart-Beat Monitor (HBM)	AM	Autonomic Manager (Stationary agent)
☺	Autonomic Agent (Mobile agent)		



AC – The Autonomic Environment extending it to fit with NASA standards …

Table 1. NASA's Risk Index Determination.[3]

Hazard severity	Hazard probability			
	Probable	Occasional	Remote	Improbable
Catastrophic	1	1	2	3
Critical	1	2	4	4
Marginal	2	3	4	5
Negligible	3	4	5	5

Autonomic Co

Key

S*	**Self-* event messages**	**AE**	**Autonomic Element (AM+MC)**
	Pulse Monitor (PBM)	**MC**	**Managed Component**
	Heart-Beat Monitor (HBM)	**AM**	**Autonomic Manager (Stationary agent)**
	Autonomic Agent (Mobile agent)		

Why Computer-Based Systems *Should* be Autonomic

Roy Sterritt
School of Computing and Mathematics,
Faculty of Engineering
University of Ulster
Northern Ireland
r.sterritt@ulster.ac.uk

Mike Hinchey
NASA Goddard Space Flight Center
Software Engineering Laboratory
Greenbelt, MD 20771
USA
michael.g.hinchey@nasa.gov

Abstract

The objective of this paper is to discuss why computer-based systems should be autonomic, where autonomicity implies self-managing, often conceptualized in terms of being self-configuring, self-healing, self-optimising, self-protecting and self-aware. We look at motivations for autonomicity, examine how more and more systems are exhibiting autonomic behavior, and finally look at future directions.

Keywords: Autonomic Computing, Autonomic Systems, Self-Managing Systems, Complexity, Total Cost of Ownership, Future Computing Paradigms.

1. Introduction

Autonomic Computing and other self-managing initiatives have emerged as a significant vision for the design of computing based systems. Their goals are the development of systems that are self-configuring, self-healing, self-protecting and self-optimizing, among other self-* properties. The ability to achieve this selfware is dependant on self-awareness and environment awareness implemented through a feedback control loop consisting of sensors and effectors within the CBS (providing the self-monitoring and self-adjusting properties) [1]. Dependability is a long-standing desirable property of all computer-based systems while complexity has become a blocking force to achieving this. The autonomic initiatives offer a means to achieve dependability while coping with complexity [2].

The ECBS workshop on Engineering of Autonomic Systems (EASe) [3][4] aims to establish autonomicity as an integral part of a CBS and explore techniques, tools, methodologies, and so on, to make this happen. The spectrum and implications are so wide that we need to reach out to other research communities to make this a reality.

The purpose of this paper is to consider why systems should be autonomic, how many more than we might first think *are* already on the evolutionary autonomic path, and to look to the future and see what more we need to consider.

2. Facing Life As It Is…

Upon launching the Autonomic Computing initiative, IBM called on the industry to face up to the ever-increasing complexity and total cost of ownership.

2.1. Business Realities

- It is estimated that companies now spend between 33% and 50% of their total cost of ownership recovering from or preparing against failures [5].
- Many of these outages, with some estimates at as high as 40%, are caused by operators themselves [6].
- 80% of expenditure on IT is spent on operations, maintenance and minor enhancements [7].
- IBM has been estimated to add about 15,000 people per year to its service organization in order to assist customers in dealing with complex platforms [8]. One can assume that other large organizations are being required to make correspondingly large expansions to keep ahead (or even to keep up).

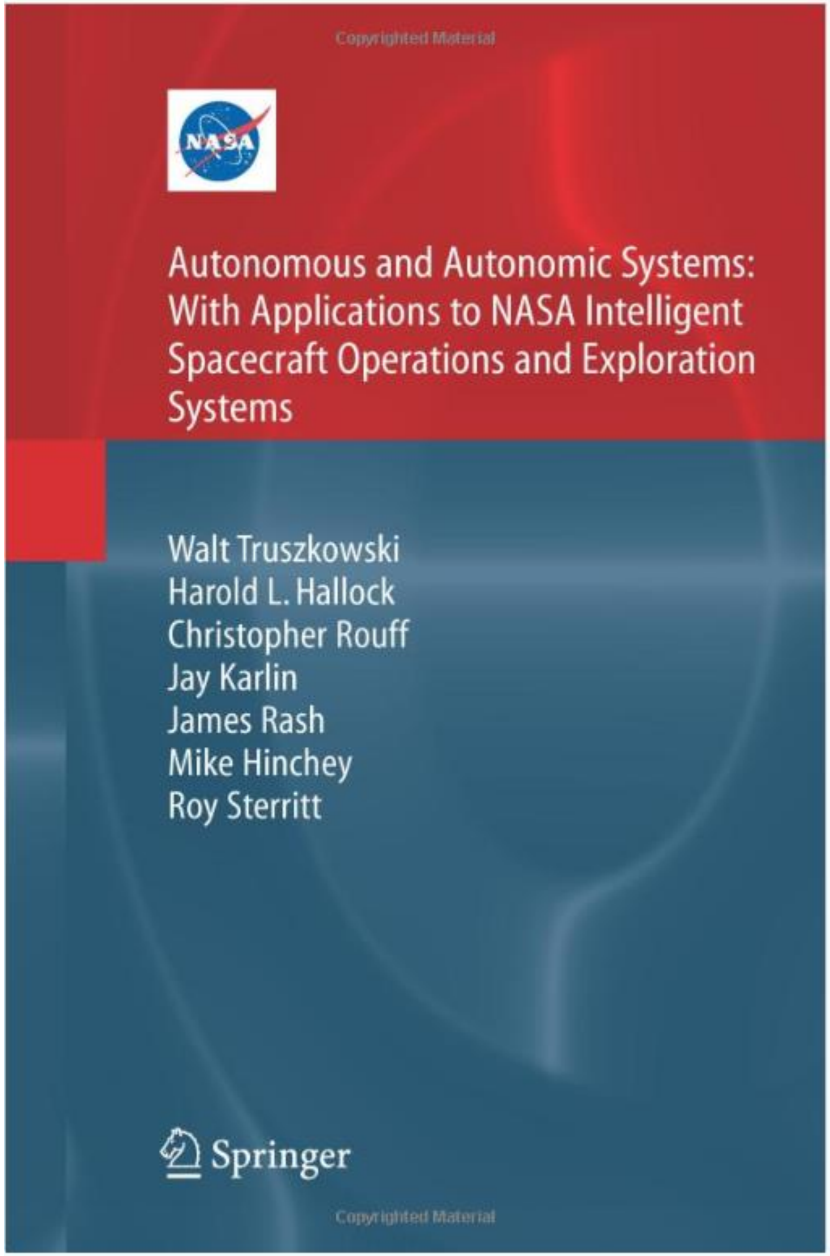
These realities together with the complexity and total cost of ownership (TCO) problem all highlight the need for a change.

2.2. Complexity

The world is becoming an ever-increasing complex place. In terms of computer systems, this complexity has been confounded by the drive towards cheaper,

Proceedings of the 12th IEEE International Conference and Workshops on the Engineering of Computer-Based Systems (ECBS'05)
0-7695-2308-0/05 $20.00 © 2005 IEEE

COMPUTER SOCIETY



Why Computer-Based Systems *Should* be Autonomic

Roy Sterritt
School of Computing and Mathematics,
Faculty of Engineering
University of Ulster
Northern Ireland
r.sterritt@ulster.ac.uk

Mike Hinchey
NASA Goddard Space Flight Center
Software Engineering Laboratory
Greenbelt, MD 20771
USA
michael.g.hinchey@nasa.gov

Abstract

The objective of this paper is to discuss why computer-based systems should be autonomic, where autonomicity implies self-managing, often conceptualized in terms of being self-configuring, self-

The purpose of this paper is to consider why systems should be autonomic, how many more than we might first think *are* already on the evolutionary autonomic path, and to look to the future and see what more we need to consider.

These précis of past, current, and future, can only lead to the conclusion that computer-based systems *should* be autonomic.

dependant on self-awareness and environment awareness implemented through a feedback control loop consisting of sensors and effectors within the CBS (providing the self-monitoring and self-adjusting properties) [1]. Dependability is a long-standing desirable property of all computer-based systems while complexity has become a blocking force to achieving this. The autonomic initiatives offer a means to achieve dependability while coping with complexity [2].

The ECBS workshop on Engineering of Autonomic Systems (EASe) [3][4] aims to establish autonomicity as an integral part of a CBS and explore techniques, tools, methodologies, and so on, to make this happen. The spectrum and implications are so wide that we need to reach out to other research communities to make this a reality.

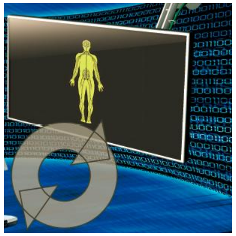
- IBM has been estimated to add about 15,000 people per year to its service organization in order to assist customers in dealing with complex platforms [8]. One can assume that other large organizations are being required to make correspondingly large expansions to keep ahead (or even to keep up).

These realities together with the complexity and total cost of ownership (TCO) problem all highlight the need for a change.

2.2. Complexity

The world is becoming an ever-increasingly complex place. In terms of computer systems, this complexity has been confounded by the drive towards cheaper,

Proceedings of the 12th IEEE International Conference and Workshops on the Engineering of Computer-Based Systems (ECBS'05)



Autonomic Computing –
Other Emerging self- properties*

- *self-anticipating,*
- *self-adapting,*
- *self-critical,*
- *self-defining,*
- *self-diagnosis,*
- *self-governing,*
- *self-installing*
- *self-organized,*
- *self-recovery,*
- *self-reflecting,*
- *self-simulation*
- *self-stabilizing*
- self-*

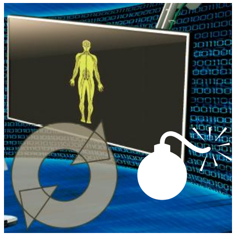
Self-managing :
(wo)man in the loop?

All positive?

Issue : Trust

- *selfish ☺ Unexpected emergent behaviour*

Race conditions etc.



Apoptosis – *returning to Biological metaphors*

- if you cut yourself and it starts bleeding…

- often, the cut will have caused skin cells to be displaced down into muscle tissue

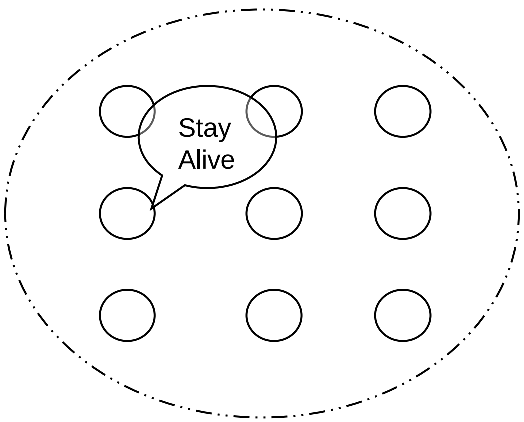
- if they survive and divide, they have the potential to grow into a tumour

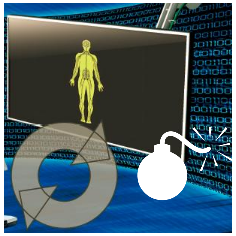
- the body's solution to dealing with this is cell self-destruction

- with mounting evidence that certain types of cancer are the result of cells not dying fast enough, rather than multiplying out of control

- It is believed that a cell knows when to commit suicide because cells are programmed to do so

- i.e. self-destruct (sD) is an intrinsic property.





Apoptosis – *returning to Biological metaphors* (cont.)

- this sD is delayed due to the continuous receipt of biochemical reprieves.

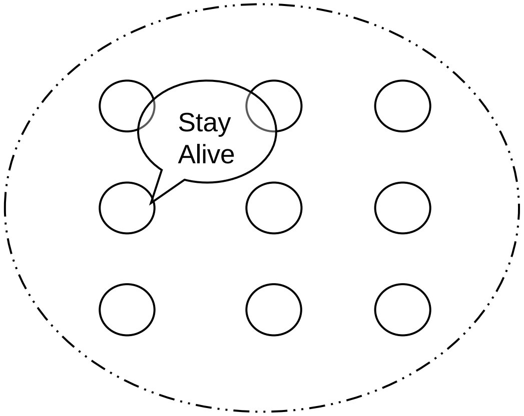
- this process is referred to as *apoptosis*,

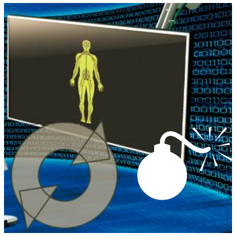
- meaning ‘drop out’,
  (used by the Greeks to refer to the Autumn dropping of leaves from trees);

- i.e., loss of cells that ought to die in the midst of the living structure.

- the process has also been nicknamed ‘death by default’,

- where cells are prevented from putting an end to themselves
  due to constant receipt of biochemical ‘stay alive’ signals





Self-protection & Trust
Security issues with Agents

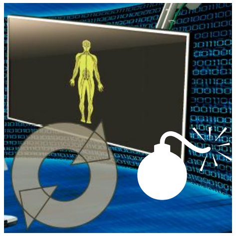
Greenburg (1998) highlighted the situation simply by recalling the situation where the server

omega.univ.edu was decommissioned

- its work moving to other machines.
- when a few years later a new computer was assigned the old name.
- to the surprise of everyone email arrived, much of it 3 years old.
- the mail had survived ‘pending’ on Internet relays waiting for omega.univ.edu to come back up.

The same situation could arise for mobile agents;
these would not be rogue mobile agents –
they would be carrying proper authenticated credentials.

The mobile autonomic agent could cause substantial damage,
e.g., deliver an archaic upgrade (self-configuration)
to part of the network operating system resulting in bringing down the entire network



Self-protection & Trust
self-destruct property a solution?

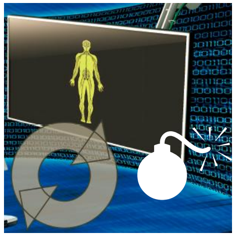
Generally the security concerns with agents

- misuse of hosts by agents (accidental)
  - accidental or unintentional situations caused by that agent (race conditions and unexpected emergent behavior)

- misuse of agents by hosts (accidental or deliberate)

- misuse of agents by other agents (accidental or deliberate)

  - the latter two through deliberate or accidental situations caused by external bodies acting upon the agent.
  - the range of these situations and attacks have been categorized as: damage, denial-of-service, breach-of-privacy, harassment, social engineering, event-triggered attacks, and compound attacks.



Self-protection & Trust
self-destruct property a solution? ***(cont.)***

In the situation where portions of an agent’s binary image e.g.,

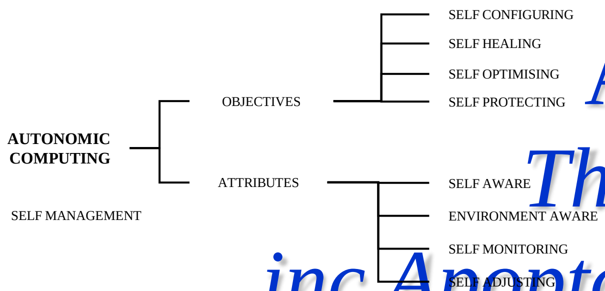
monetary certificates,
keys,
information, etc.

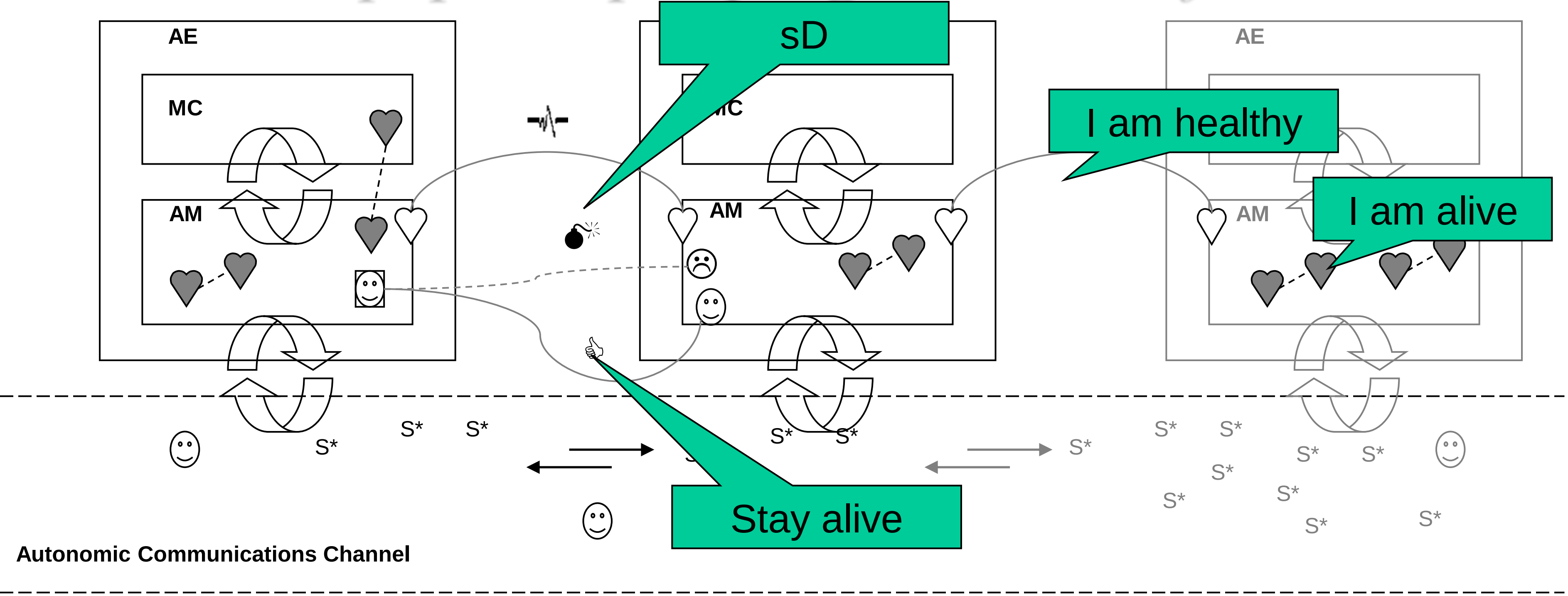
are vulnerable to being copied when visiting a host,
this can be prevented by encryption.

Yet there has to be decryption in order to execute,
which provides a window of vulnerability.

This situation has similar overtones to biological apoptosis,
where the body is at its most vulnerable during cell division.

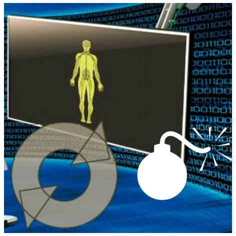
Autonomic Computing –
The Autonomic Environment
inc Apoptotic pre-programmed self-destruct



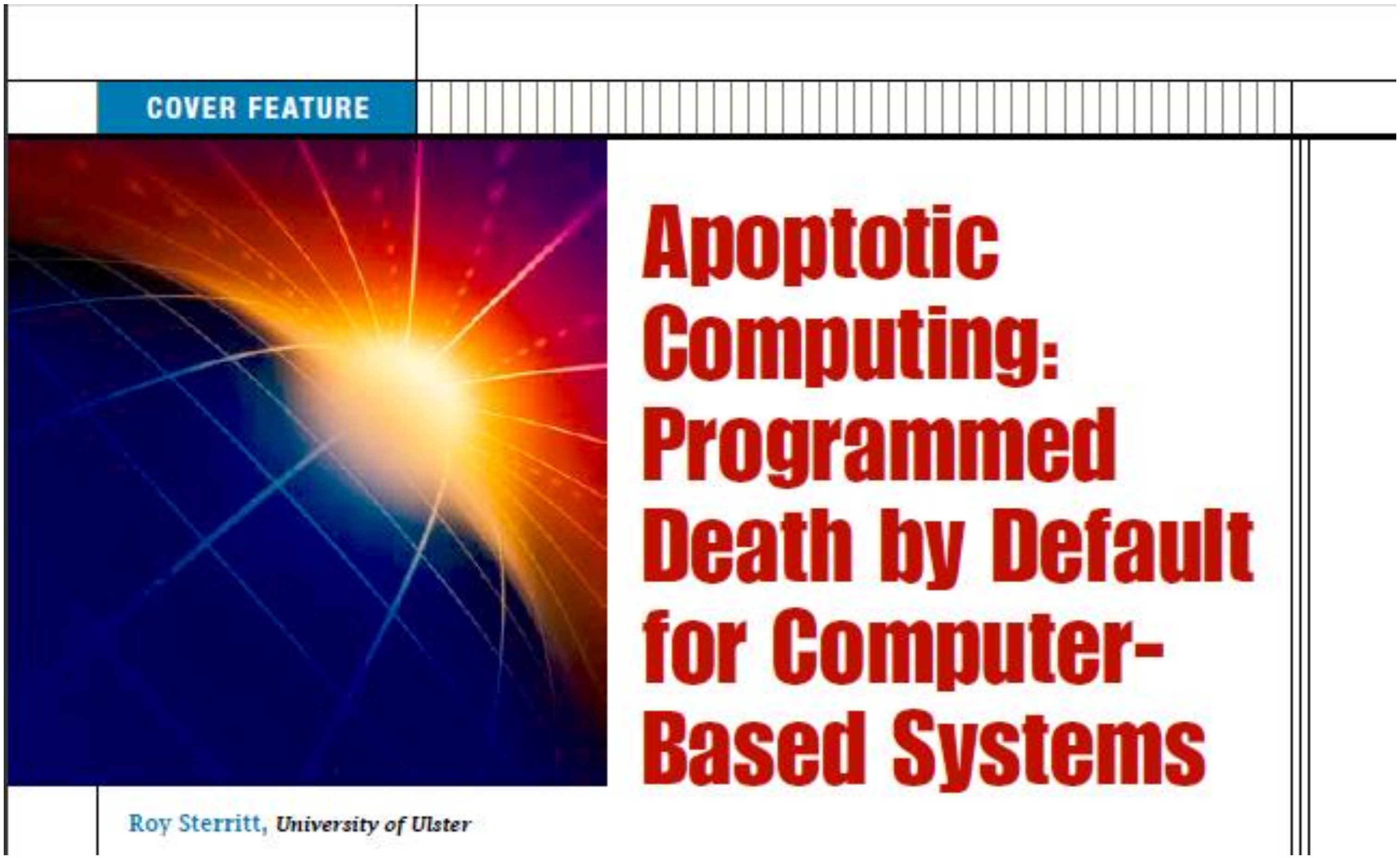


Key

S*	**Self-* event messages**	**AE**	**Autonomic Element (AM+MC)**
	Pulse Monitor (PBM)	**MC**	**Managed Component**
	Heart-Beat Monitor (HBM)	**AM**	**Autonomic Manager (Stationary agent)**
	Autonomic Agent (Mobile agent)		**Autonomic Agent Apoptosis Controls**



=> Apoptotic Computing …



COVER FEATURE

Apoptotic Computing: Programmed Death by Default for Computer-Based Systems

Roy Sterritt, *University of Ulster*

[IEEE Comp. Jan 2011]

RECAP

NASA ANTS Concept Mission

- Future space missions will require cooperation between multiple satellites/rovers/craft for e.g.

ANTS (Autonomous Nano-Technology Swarm)

- Developers are proposing intelligent, autonomous swarms to do new science

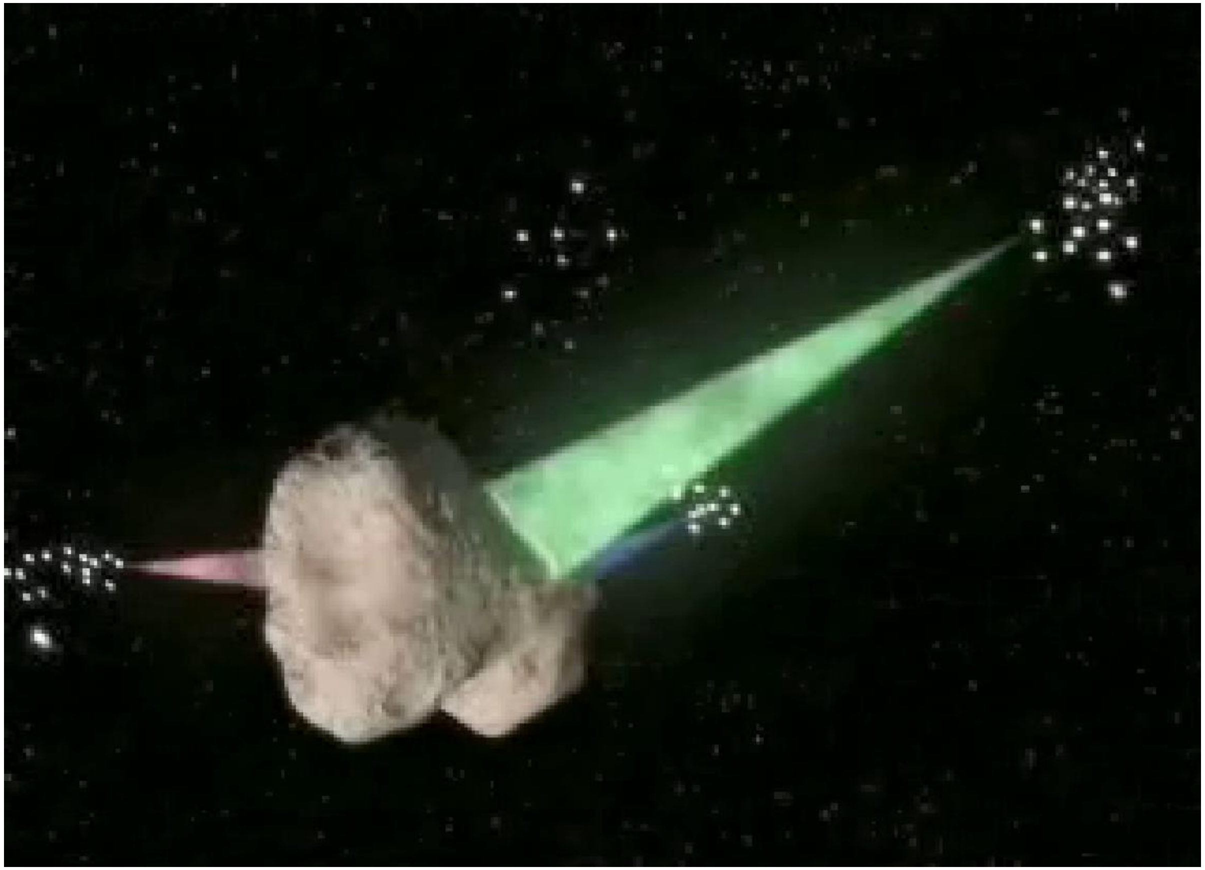
- Swarm-based systems are highly parallel and nondeterministic

- Testing these systems using current techniques will be difficult to impossible

- This raises issues for self-protection of system (mission) goals,

Curtis et al 2000; Clark et al 2001, …
https://attic.gsfc.nasa.gov/ants/

Self-Directed & Self-Managed Exploration



Curtis et al 2000; Clark et al 2001, …
https://attic.gsfc.nasa.gov/ants/

ANTS Space Exploration Missions

- The ANTS architecture is itself inspired by biological low level social insect colonies with their success in the division of labor.
- Within their specialties, individual specialists generally outperform generalists, and with sufficiently efficient social interaction and coordination, the group of specialists generally outperforms the group of generalists.
- Thus systems designed as ANTS are built from potentially very large numbers of highly autonomous, yet socially interactive, elements.
- The architecture is self-similar in that elements and sub-elements of the system may also be recursively structured as ANTS

Curtis et al 2000; Clark et al 2001, …

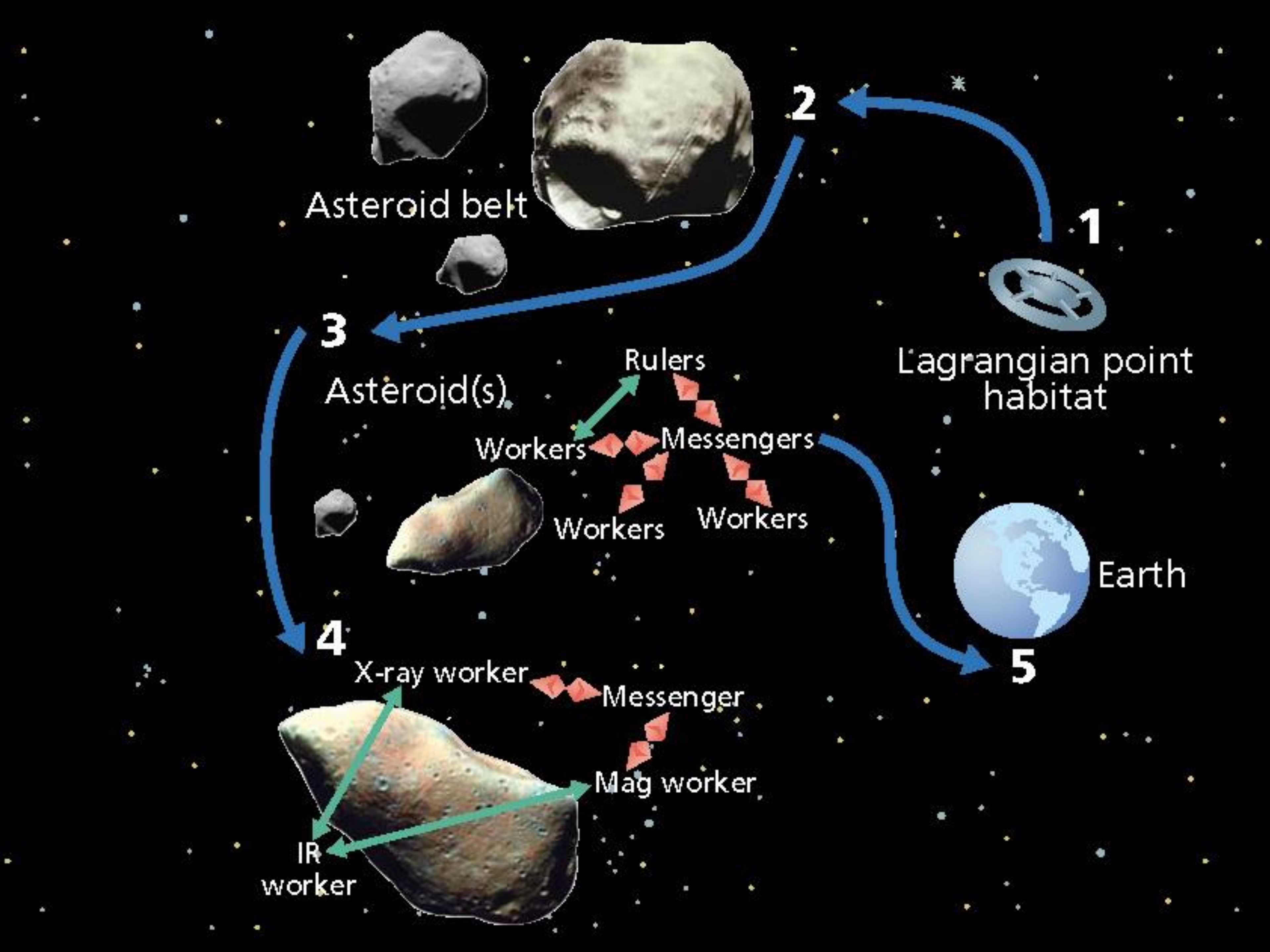
https://attic.gsfc.nasa.gov/ants/

ANTS Space Exploration Missions *(cont.)*

- Targets for ANTS-like missions include surveys of extreme environments on the
  - Earth,
  - Moon,
  - Mars, as well as
  - asteroid,
  - comet,
  - or dust populations.
- The ANTS paradigm makes the achievement of such goals possible through the use of many small, autonomous, reconfigurable, redundant element craft acting as independent or collective agents

  Curtis et al 2000; Clark et al 2001, …

  https://attic.gsfc.nasa.gov/ants/

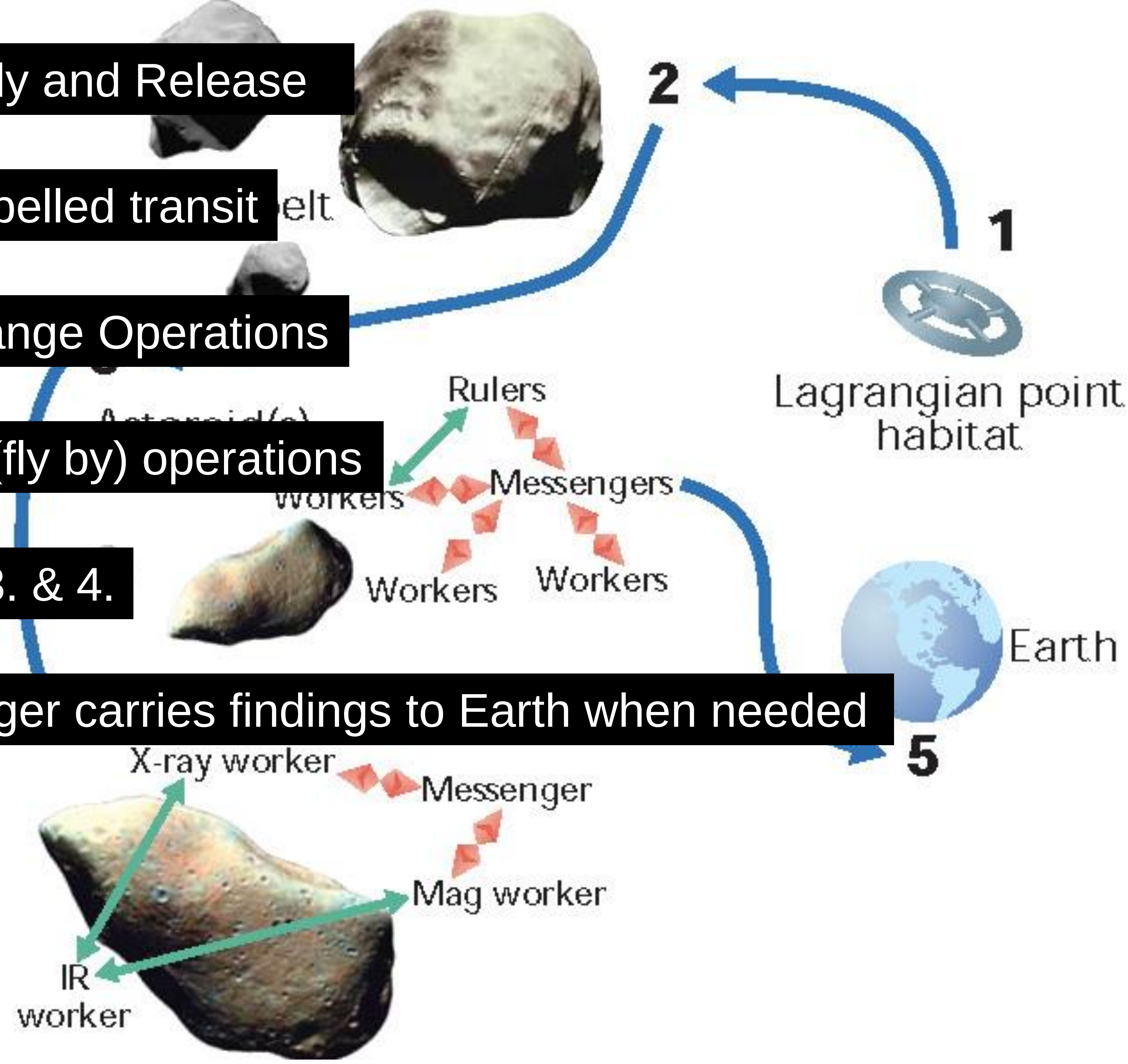


2
Asteroid belt
1
Lagrangian point habitat
3
Asteroid(s)
Rulers
Workers
Messengers
Workers
Workers
Earth
4
X-ray worker
Messenger
Mag worker
IR worker
5

1. Assembly and Release
2. Self propelled transit
3. Long Range Operations
4. Swarm (fly by) operations

Repeat 3. & 4.

5. Messenger carries findings to Earth when needed

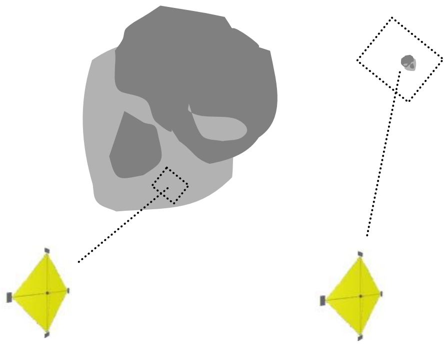


Autonomic ANTS - Self-CHOP

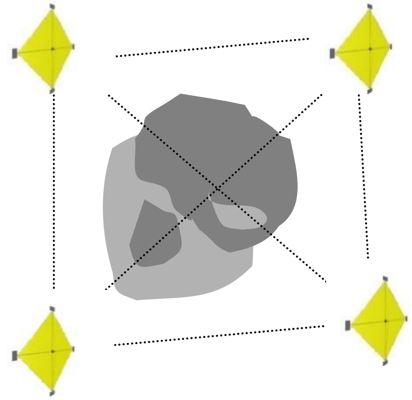
Self-Optimizing

Autonomous, Optimized Science Operations

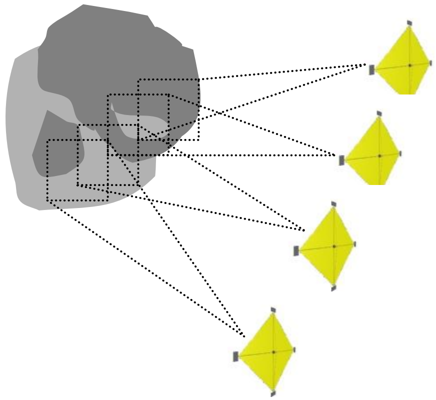
Single S/C, Local Scope
e.g. X-Ray Spectrometry
& Long-range imaging



Multi-S/C, Global Scope
e.g. Radio Science Gravimetry
“Ad hoc GPS for Asteroids”



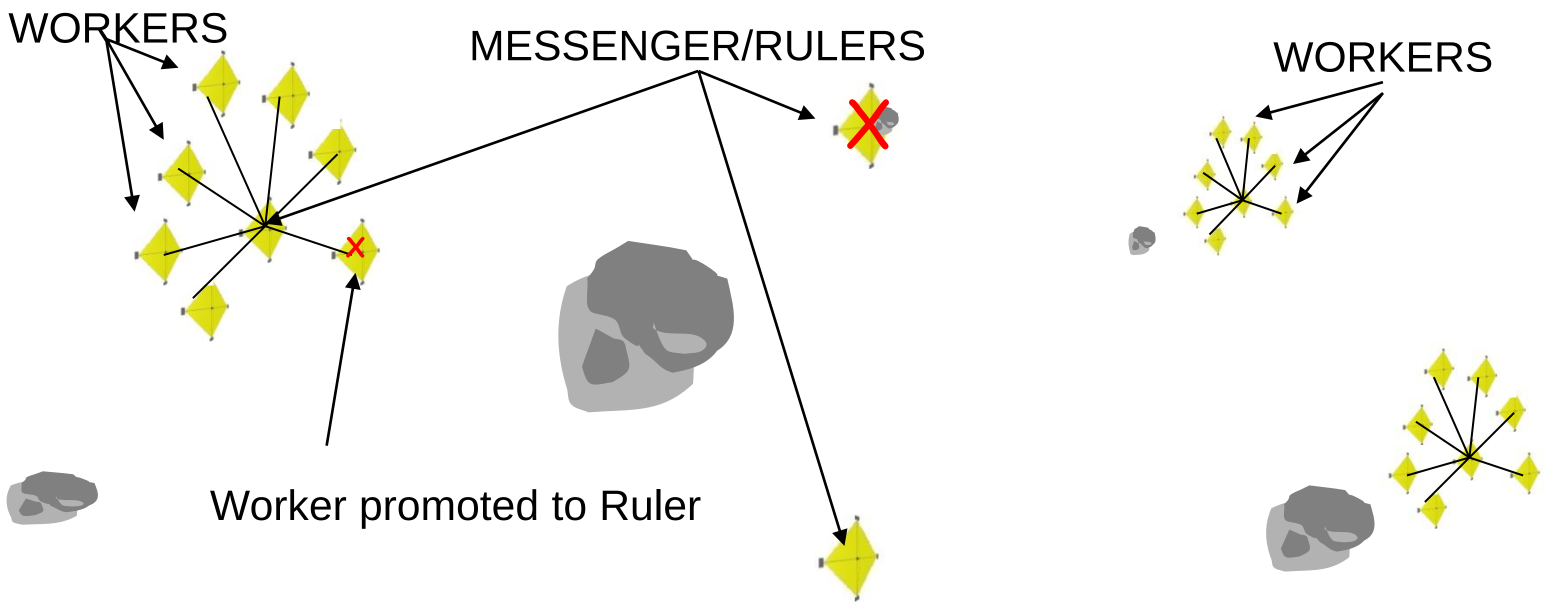
Multi-S/C, Local Scope
e.g. Imaging, Sounding,
Mapping



Autonomic ANTS Self-CHOP

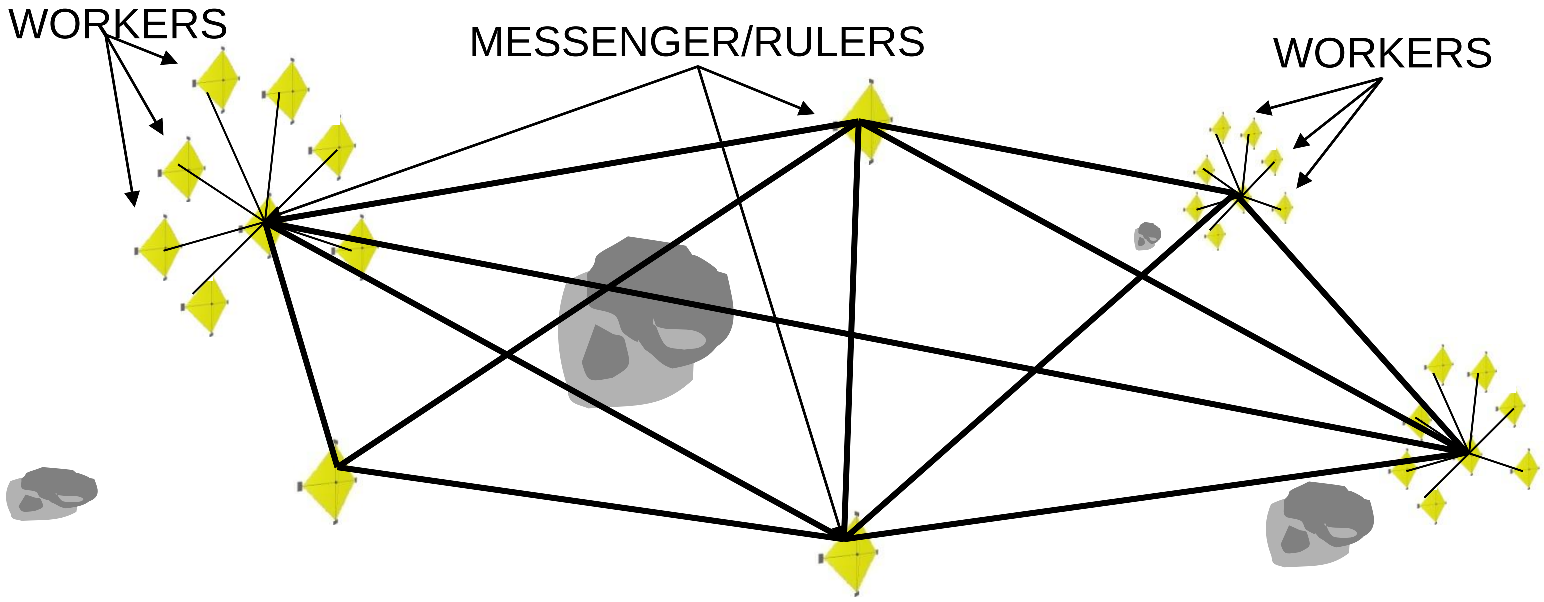
Self-Healing

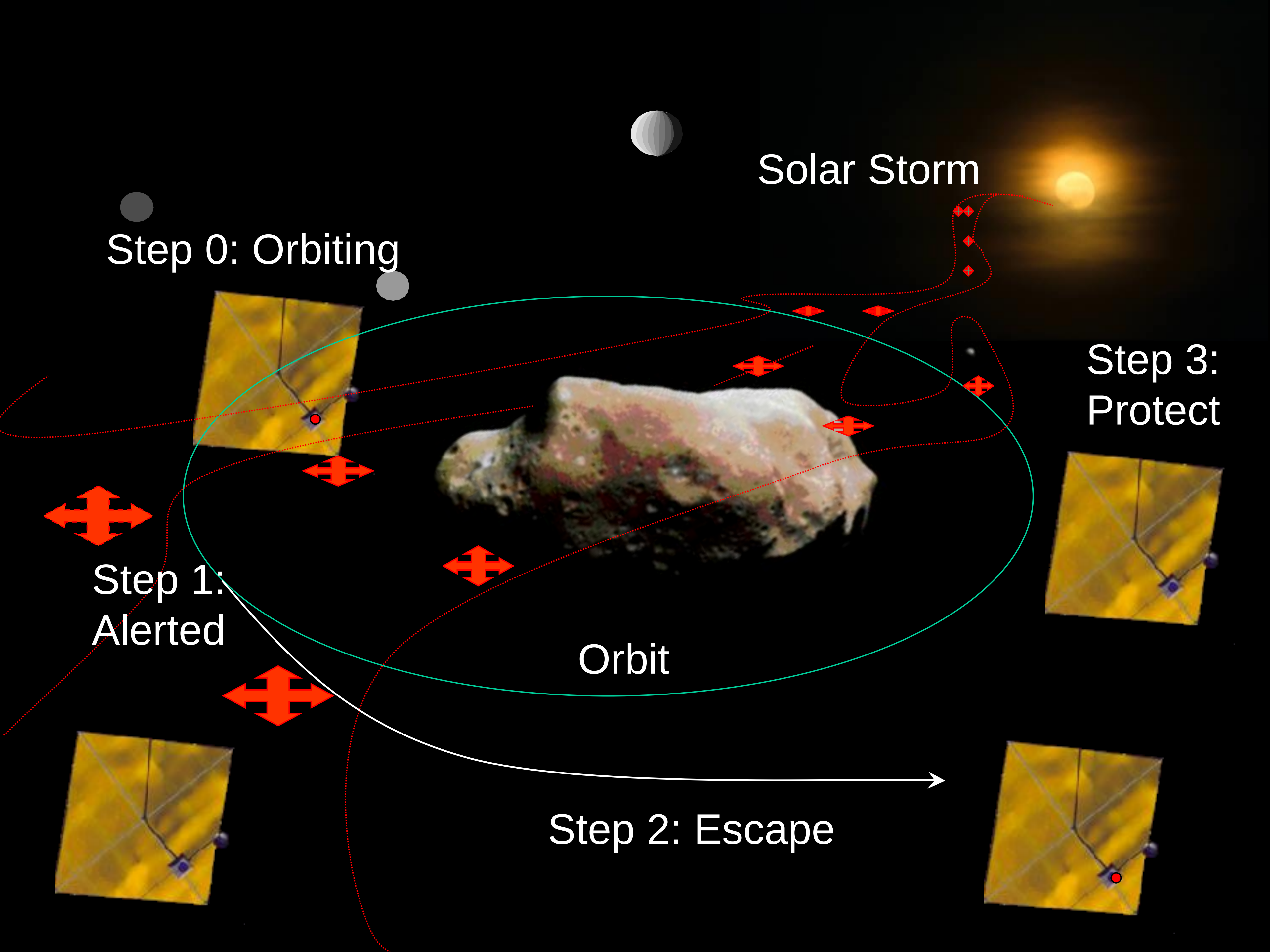
Worker instrument damaged?
Available to replace a lost Ruler or Messenger



Autonomic ANTS - Self-CHOP

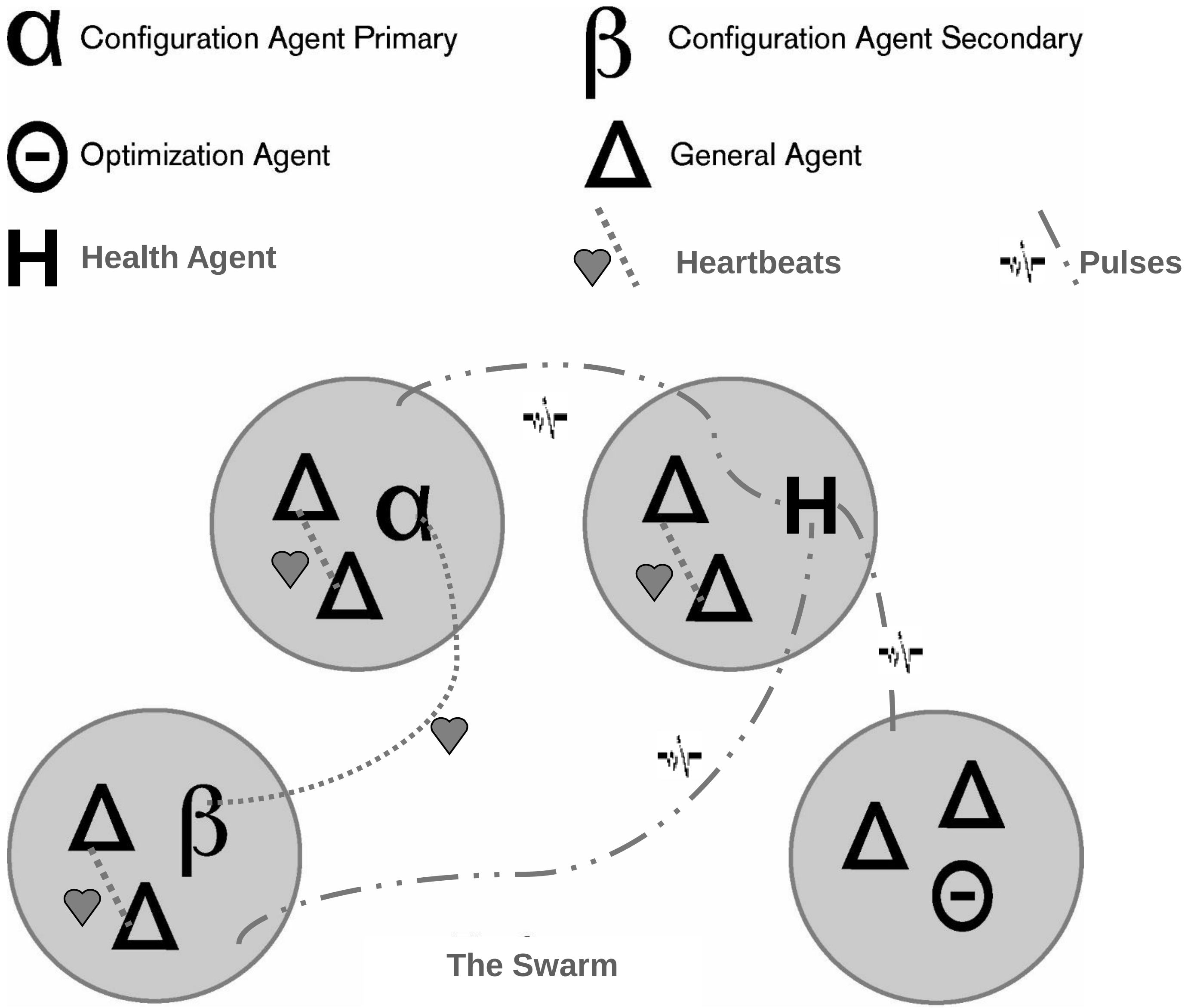
Solar Storm –Replan trajorectories or even sleep mode





Solar Storm
Step 0: Orbiting
Step 3:
Protect
Step 1:
Alerted
Orbit
Step 2: Escape

Autonomic Management - Agent Software



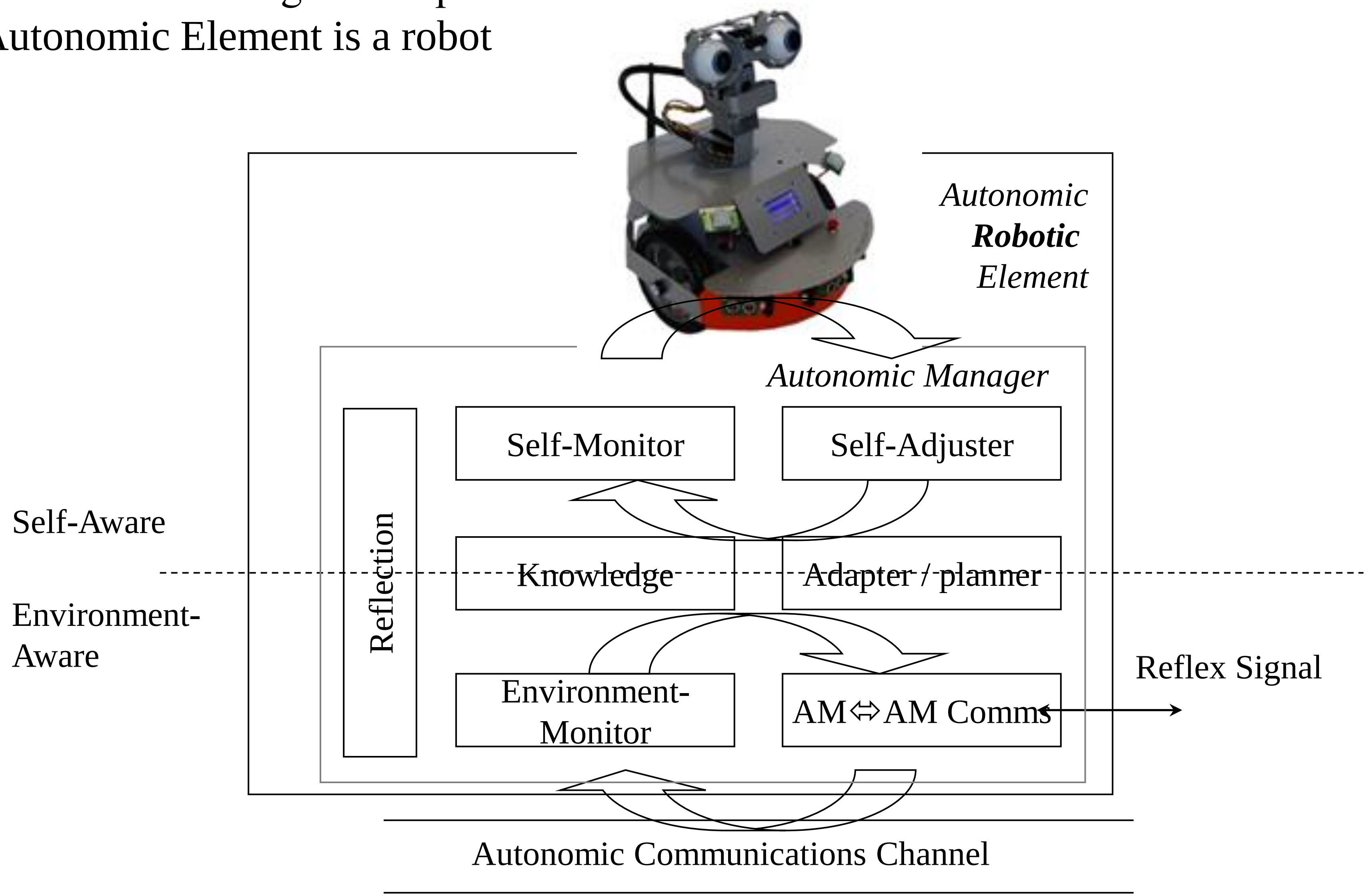
WF Truszkowski, MG Hinchey, and R Sterritt 2005

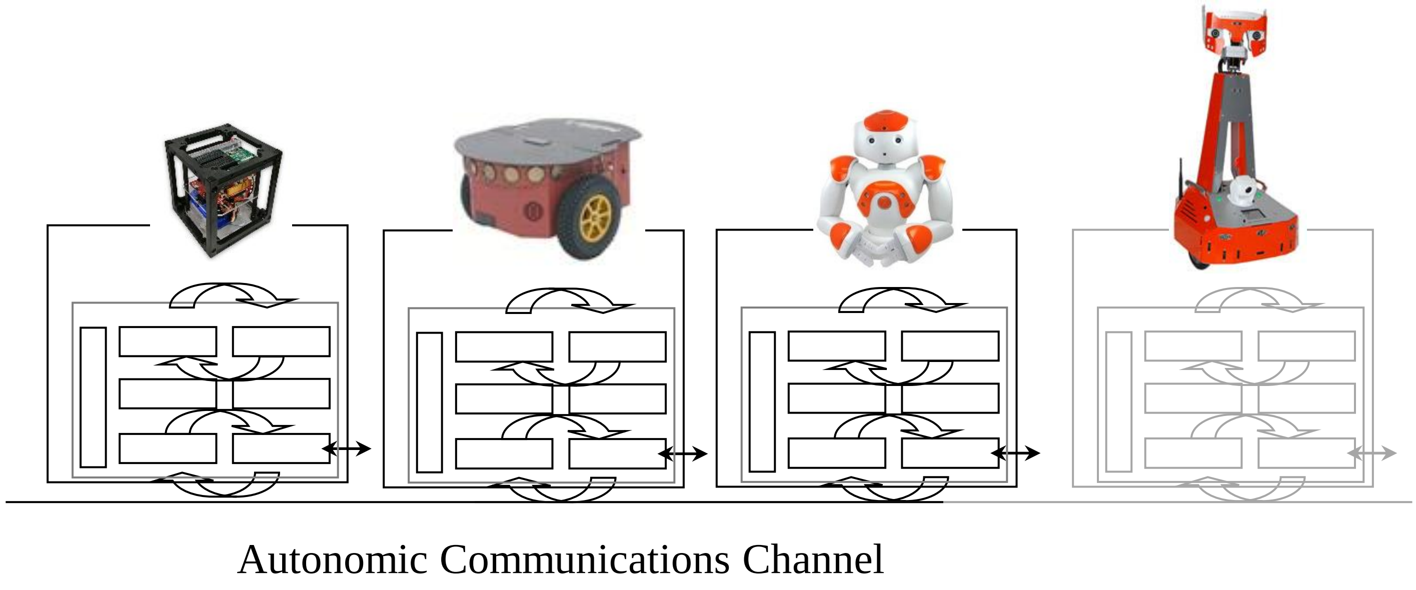
2010's Moving on from the Autonomic vision towards trustworthy systems via Autonomic & Apoptotic Computing

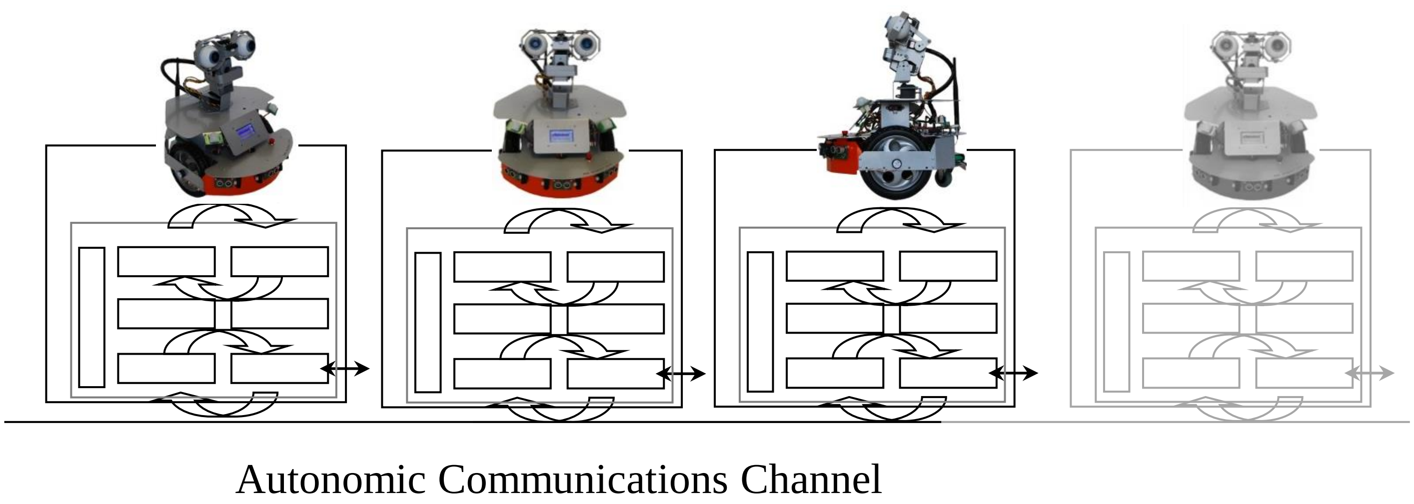
Ulster University

Autonomic Robotics

Where the "Managed Component" in the Autonomic Element is a robot




Autonomic Communications Channel


Autonomic Communications Channel

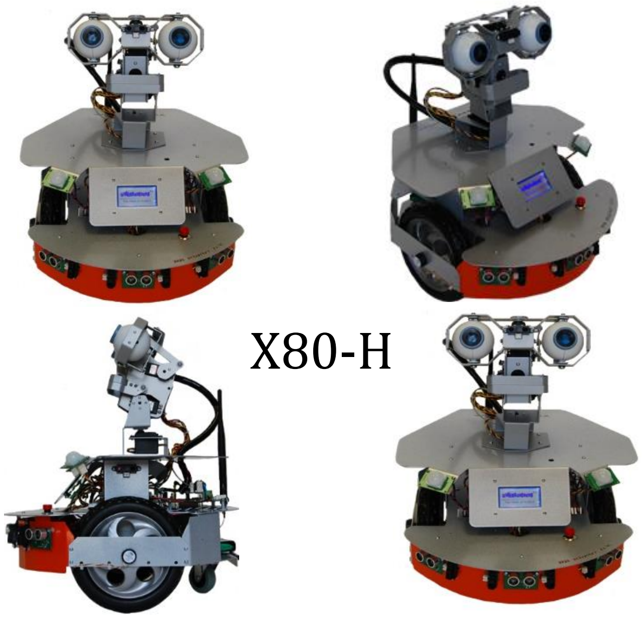
Spaces: self-properties autonomic computing for exploring spaces

- Purpose of this PhD research programme was to investigate co-operation strategies that will enable Autonomic Paradigm/Protocol change; i.e. self-recognition (of the need) and then self-configuration from swarms to constellations to clusters of robots to collaborate to perform their mission without human involvement.

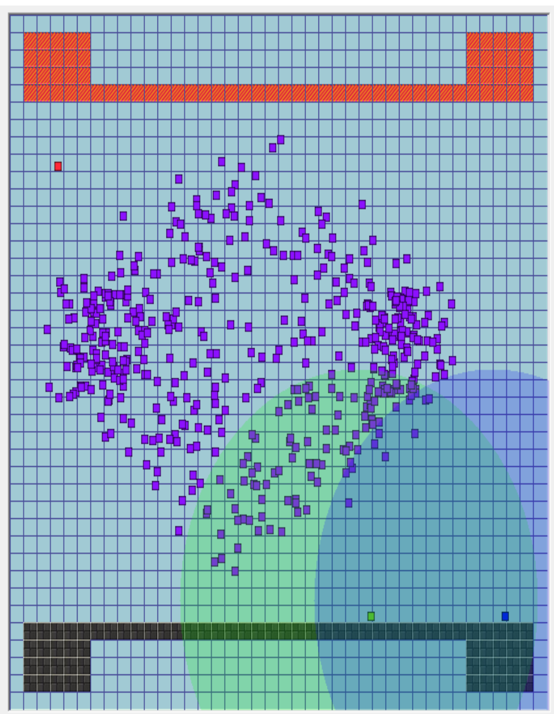
- Early research explored using 4x X80-H robots – *self-similar robots*

- Later research moved to the creation of software simulation program to implement co-operation strategies which can operate with 1000s softbots.

PhD candidate: Catherine Saunders

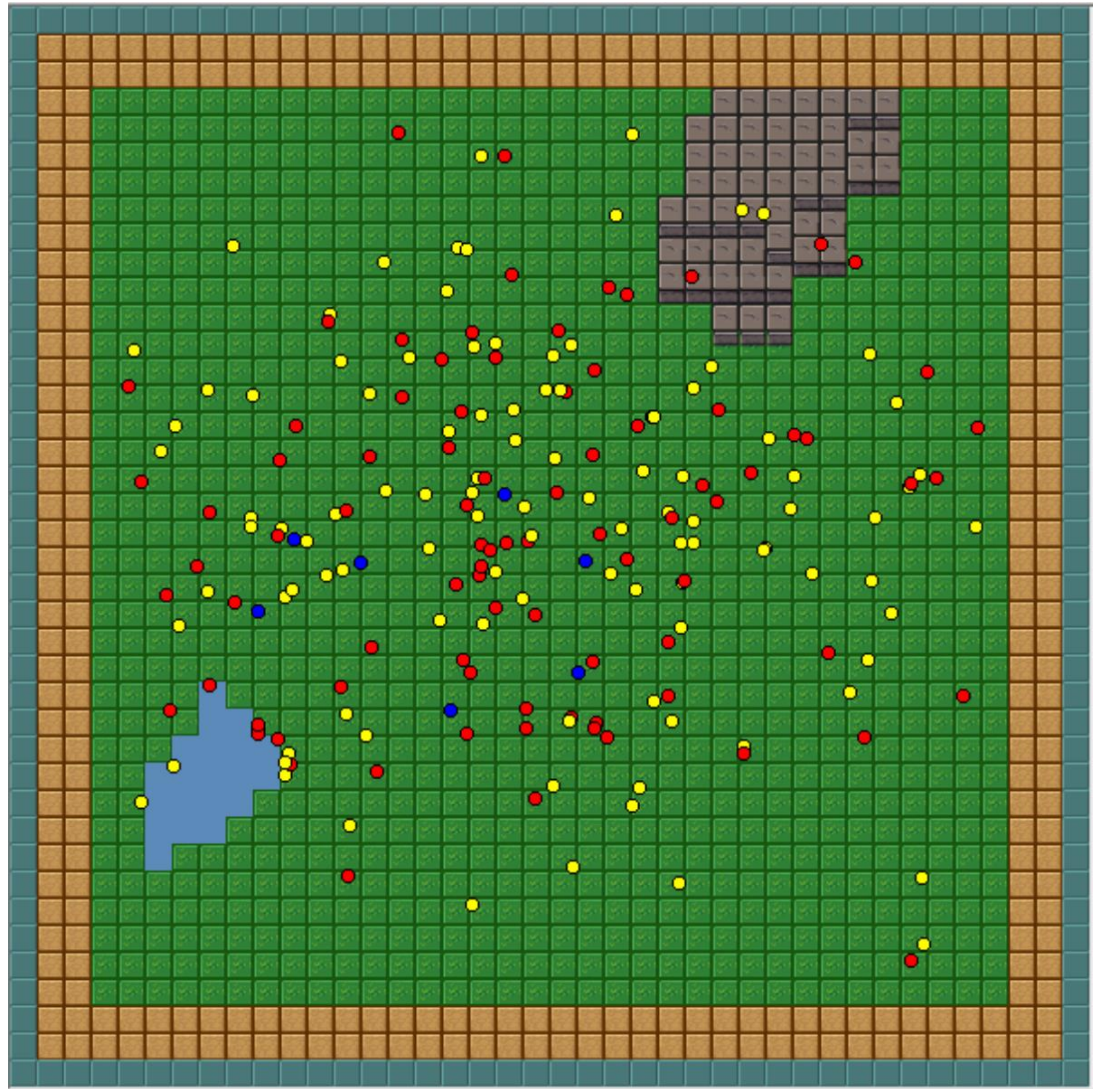


X80-H



Ruler robots and Swarm Sim.

Swarm and Message Board simulation



More information …

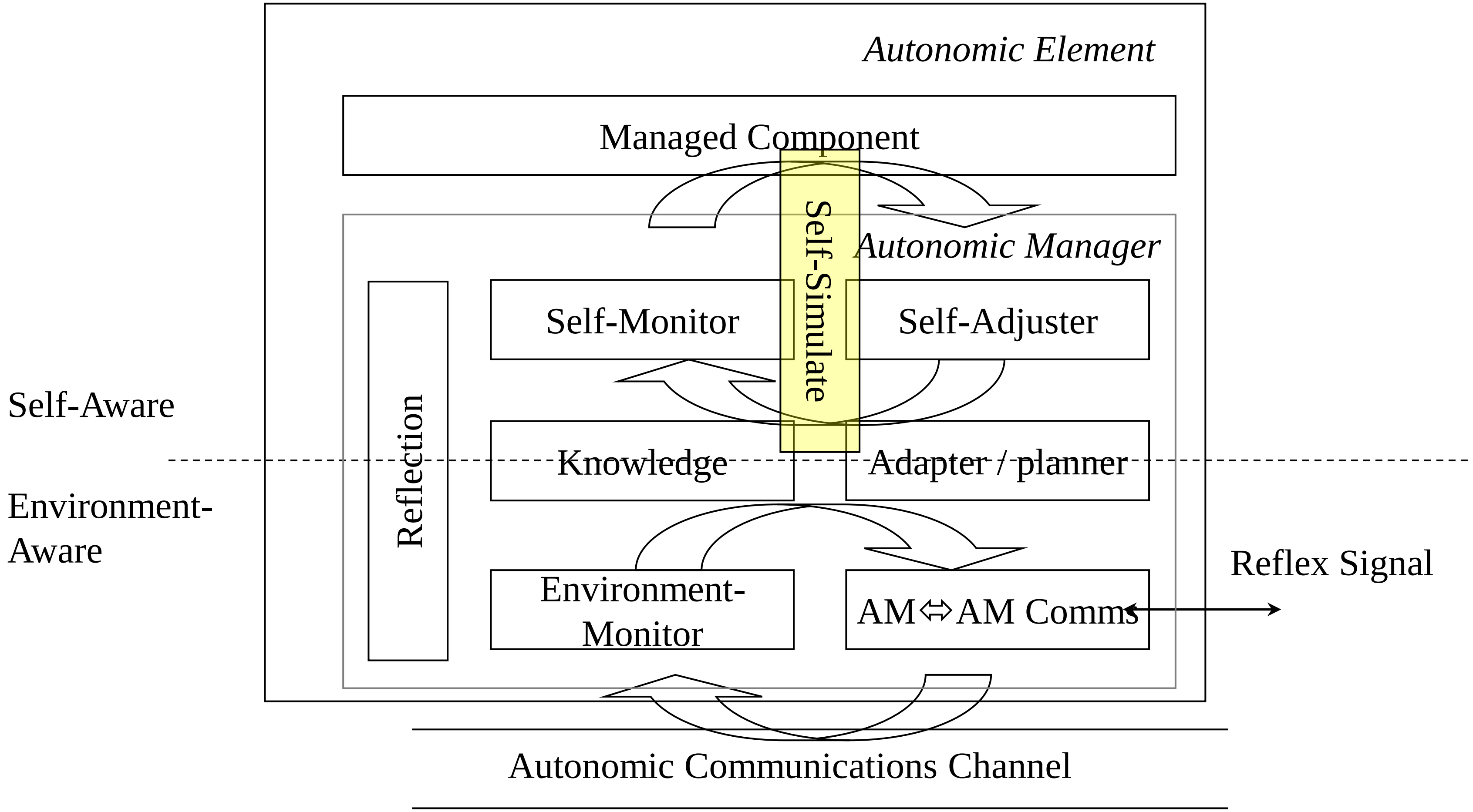
Saunders, C., Sterritt, R., & Wilkie, FG. (2020).
Collective Communication Strategies for Space Exploration.
Journal of the British Interplanetary Society, 72(12), 416. [Link]

Saunders, C., Sterritt, R., & Wilkie, G. (2016).
Autonomic Cooperation Strategies for Robot Swarms.
Adaptive 2016: The Eighth International Conference on Adaptive and Self-Adaptive Systems and Applications - Rome, Italy, Mar 2016 [Link]

Saunders, C., Sterritt, R., & Wilkie, G. (2015).
Computer Vision Techniques for Autonomic Collaboration between Mobile Robots
7th International Conference on Adaptive and Self-Adaptive Systems and Applications - Nice, France, Mar 2015 [Link]

Saunders, C., Sterritt, R., & Wilkie, G. (2014).
The Utility of Robot Sensory Devices in a Collaborative Autonomic Environment
11th IEEE International Conference and Workshops on the Engineering of Autonomic & Autonomous Sept. 2014 [Link]

Extension / Contribution to Autonomic Systems



And …

Autonomic Paradigm/Protocol change

STAAR: Self-targeting autonomic and apoptotic responses

- Purpose of this PhD research programme was to investigate self-* healing strategies and to determine if 3 tier Intelligent Machine Architecture had something to add to the MAPE-K architecture for Autonomic Robotics.

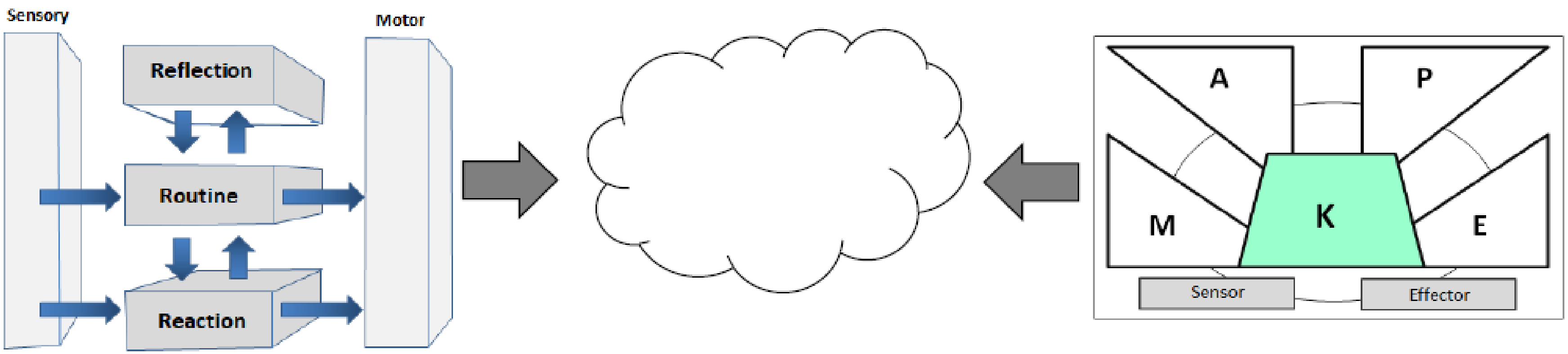
- Case studies carried out on X80-H & Pioneer robots

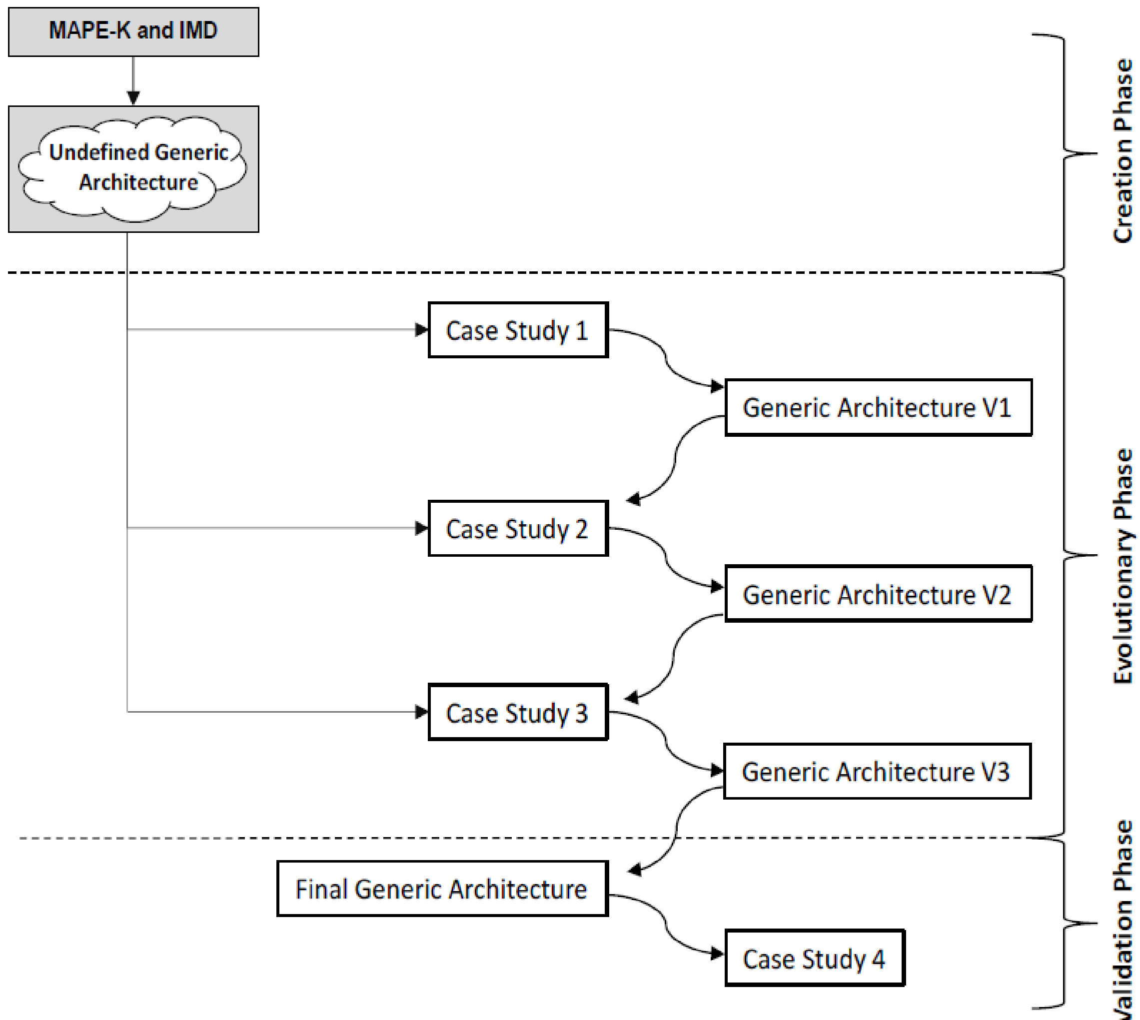
- Generalised cases studies to develop a *Generic Architecture for Autonomic Fault Handling in Mobile Robots*

PhD candidate: Martin Doran

Autonomic Robotics Architecture

IMD Architecture **AIFH Architecture** **MAPE-K Architecture**



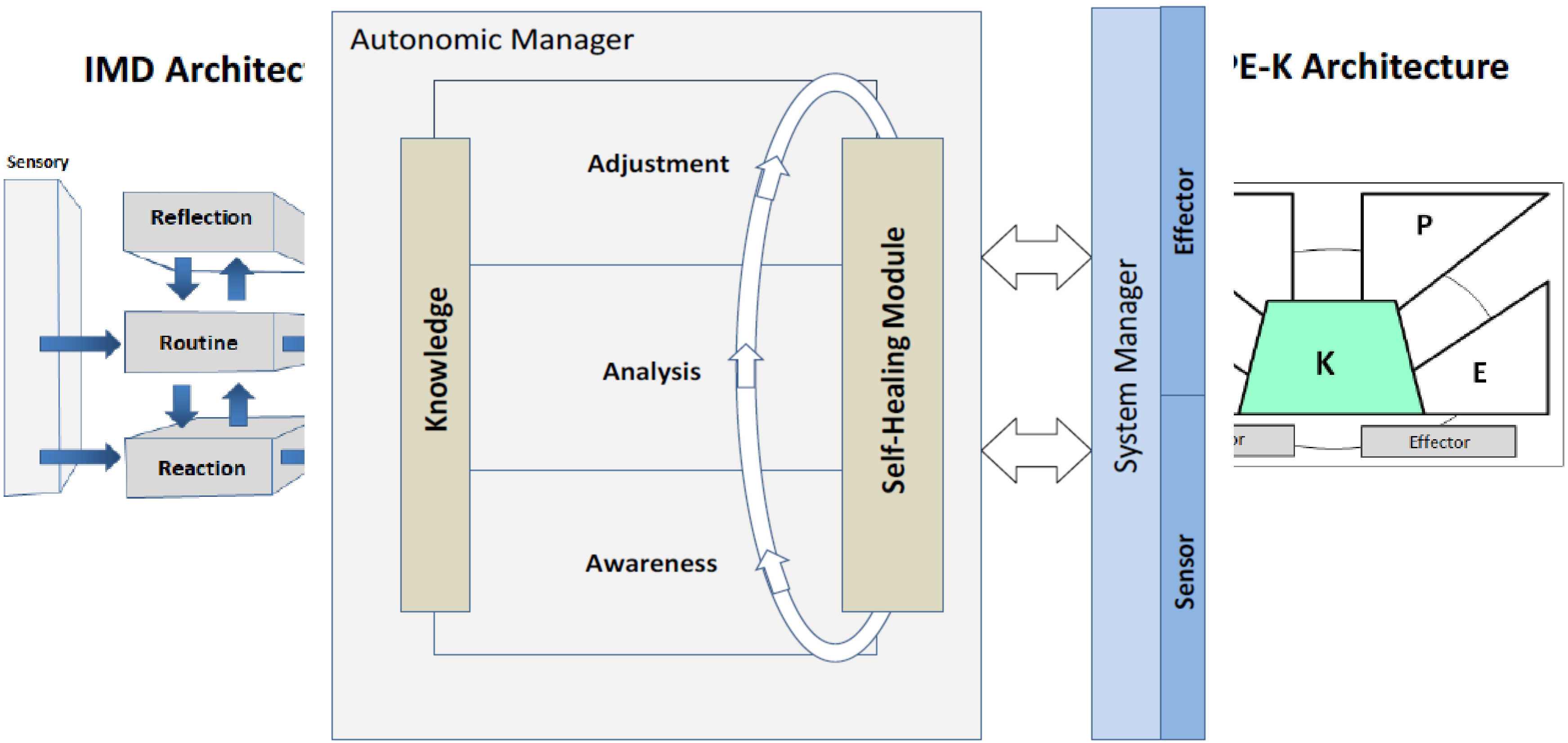


MAPE-K and IMD
Undefined Generic Architecture
Case Study 1
Generic Architecture V1
Case Study 2
Generic Architecture V2
Case Study 3
Generic Architecture V3
Final Generic Architecture
Case Study 4
Creation Phase
Evolutionary Phase
Validation Phase

Case Studies Published

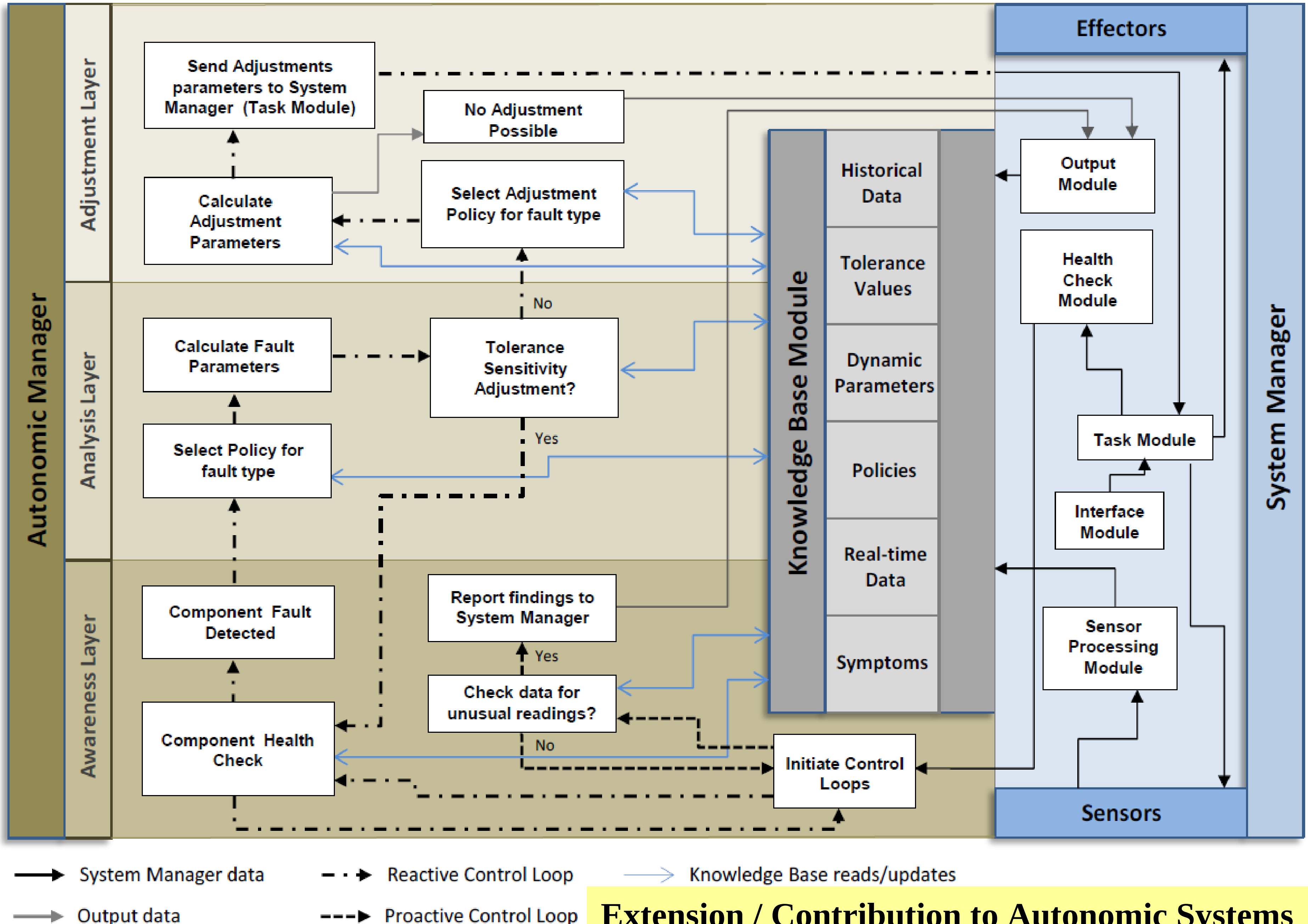
- Doran, M., Sterritt, R., & Wilkie, G. (2020), **Autonomic Architecture for Fault Handling in Mobile Robots**.
Innovations in System and Software Engineering, NASA Journal, Springer Publications [Link] [Pub-link]
- Doran, M., Sterritt, R., & Wilkie, G. (2018).
Autonomic Management for Mobile Robot Battery Degradation.
ICACCE 2018 : 20th International Conference on Autonomic Computing and Computer Engineering - London, UK, May 2018 [Link]
- Doran, M., Sterritt, R., & Wilkie, G. (2017).
Autonomic Sonar Sensor Fault Manager for Mobile Robots.
ICACCE 2017 : 19th International Conference on Autonomic Computing and Computer Engineering London, UK, Mar 2017 [Link]
- Doran, M., Sterritt, R., & Wilkie, G. (2016).
Autonomic Self-Adaptive Robot Wheel Alignment.
Adaptive 2016: The Eighth International Conference on Adaptive and Self-Adaptive Systems and Applications (pp. 27-33) [Link]

Autonomic Robotics Architecture



Extension / Contribution to Autonomic Systems

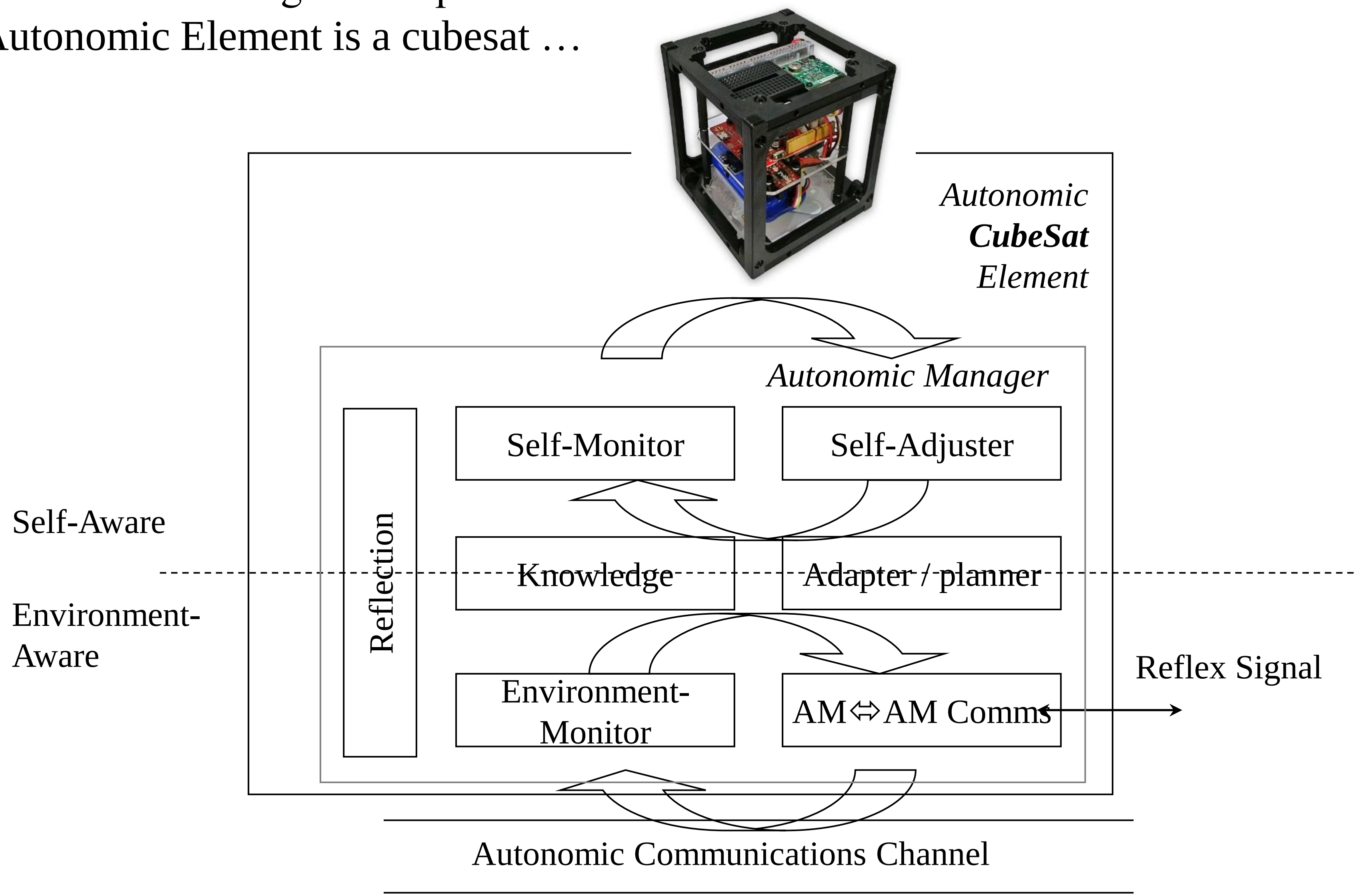
Generic Architecture for Fault Detection (AIFH)



Extension / Contribution to Autonomic Systems

Autonomic CubeSats

Where the “Managed Component” in the Autonomic Element is a cubesat …



SATURN: Swarms, Autonomic Technologies, Ubiquitous Robotics & Nanotech software paradigms for future space exploration

- Purpose of this PhD research programme is to investigate if/how the Autonomic Computing paradigm could contribute to CubeSats.

- Early research explored carrying on from Palmer Autonomic & Apoptotic demo on Arduinos with DemoSat equipment

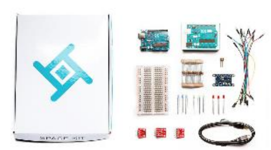
- Later research moved towards defining a ***Cubesat Autonomicity Capability Model*** (as a roadmap for future autonomic cubesat development) and an exemplar "killswitch" (Apoptotic Computing) application to prevent the CubeSat becoming space junk.

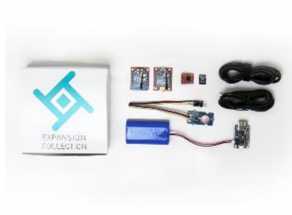
PhD candidate: Clement Gama

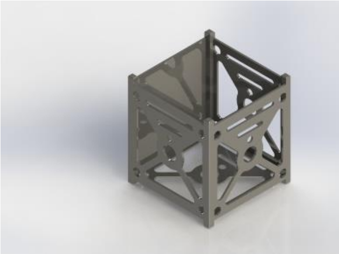
Gama, C., Sterritt, R., Wilkie, G., & Hawe, G. (2018). Towards a Cubesat Autonomicity Capability Model A Roadmap for Autonomicity in Cubesats. Proc. The Tenth International Conference on Adaptive and Self-Adaptive Systems and Applications (ADAPTIVE 2018)[Link]

Cubesats – Pre-built DemoSats

ArduSat DemoSat

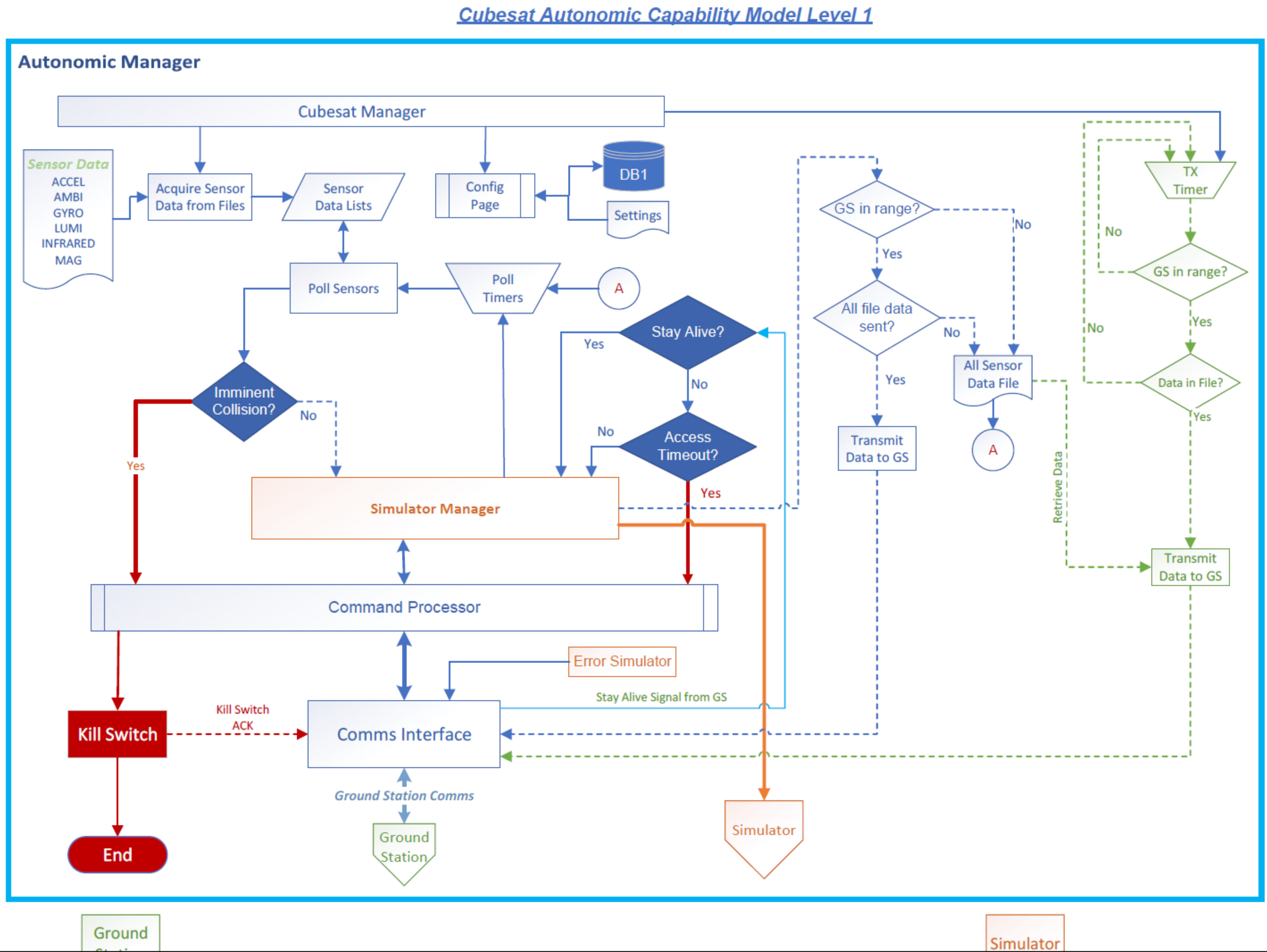






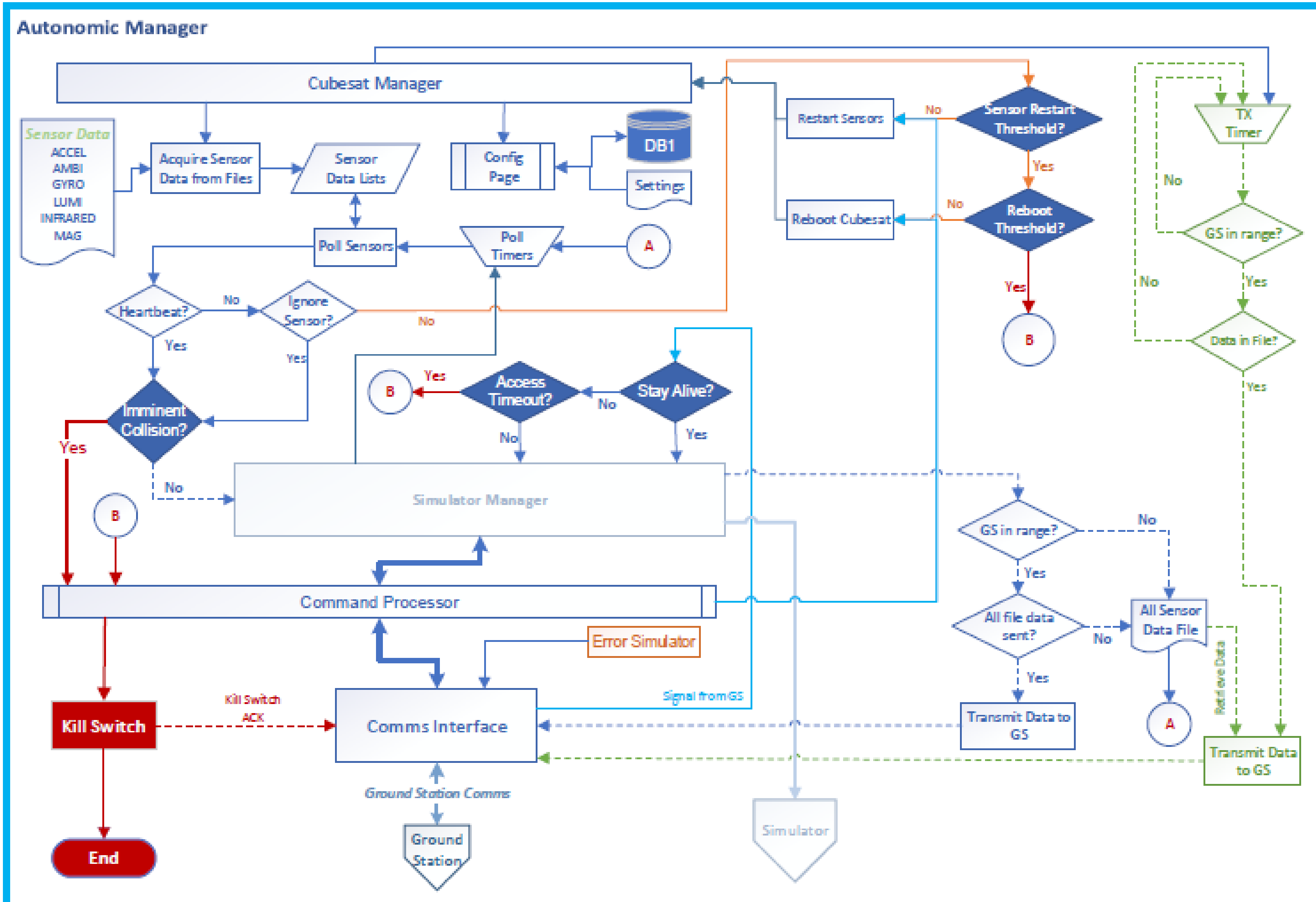
DemoSat = Spacekit + Expansion Collection + Spaceboard + Frame

CubeSat Autonomic Capability Level 1 (CubeSat)



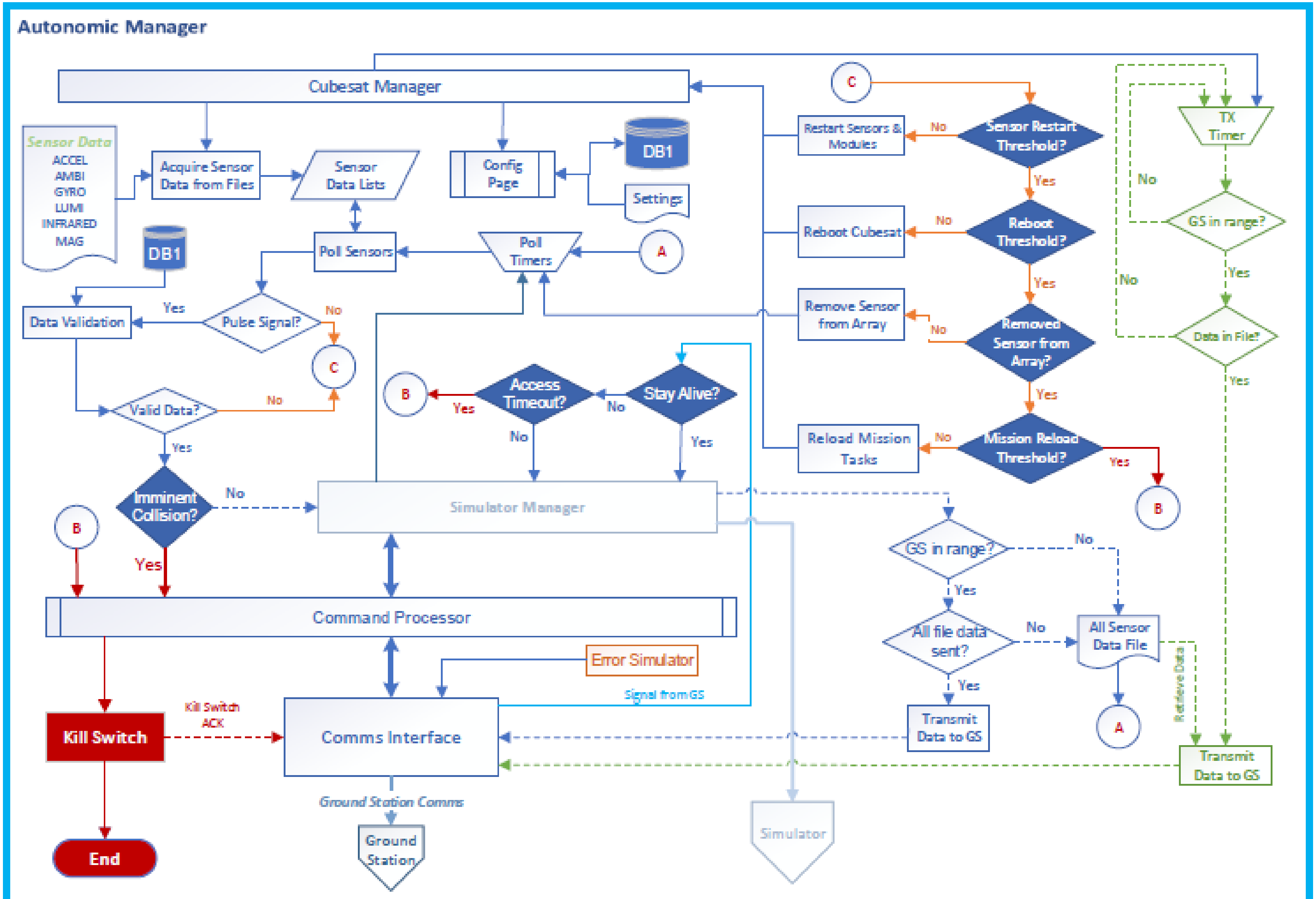
CubeSat Autonomic Capability Level 2 (CubeSat)

Cubesat Autonomic Capability Model Level 2



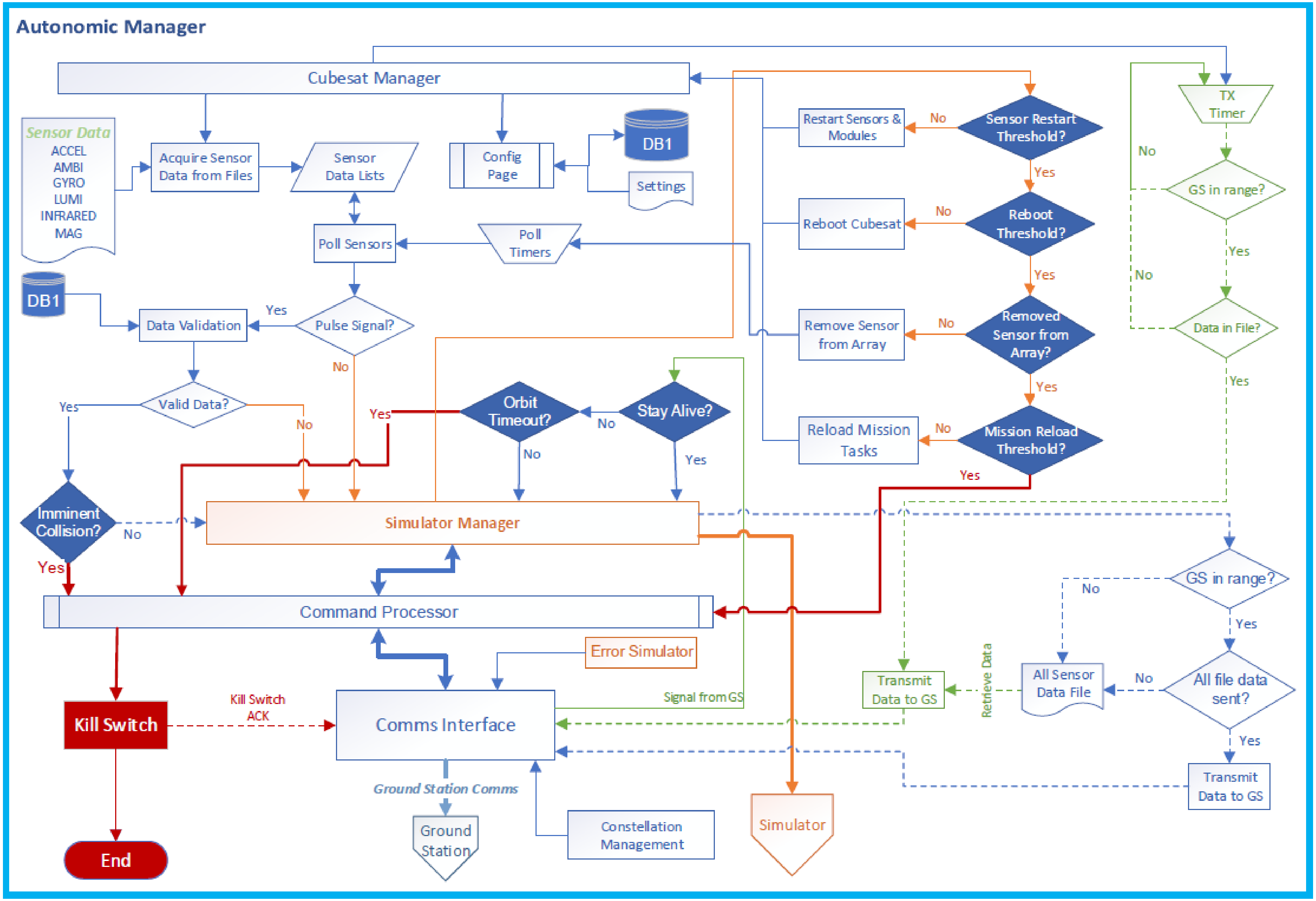
CubeSat Autonomic Capability Level 3 (CubeSat)

Cubesat Autonomic Capability Model Level 3

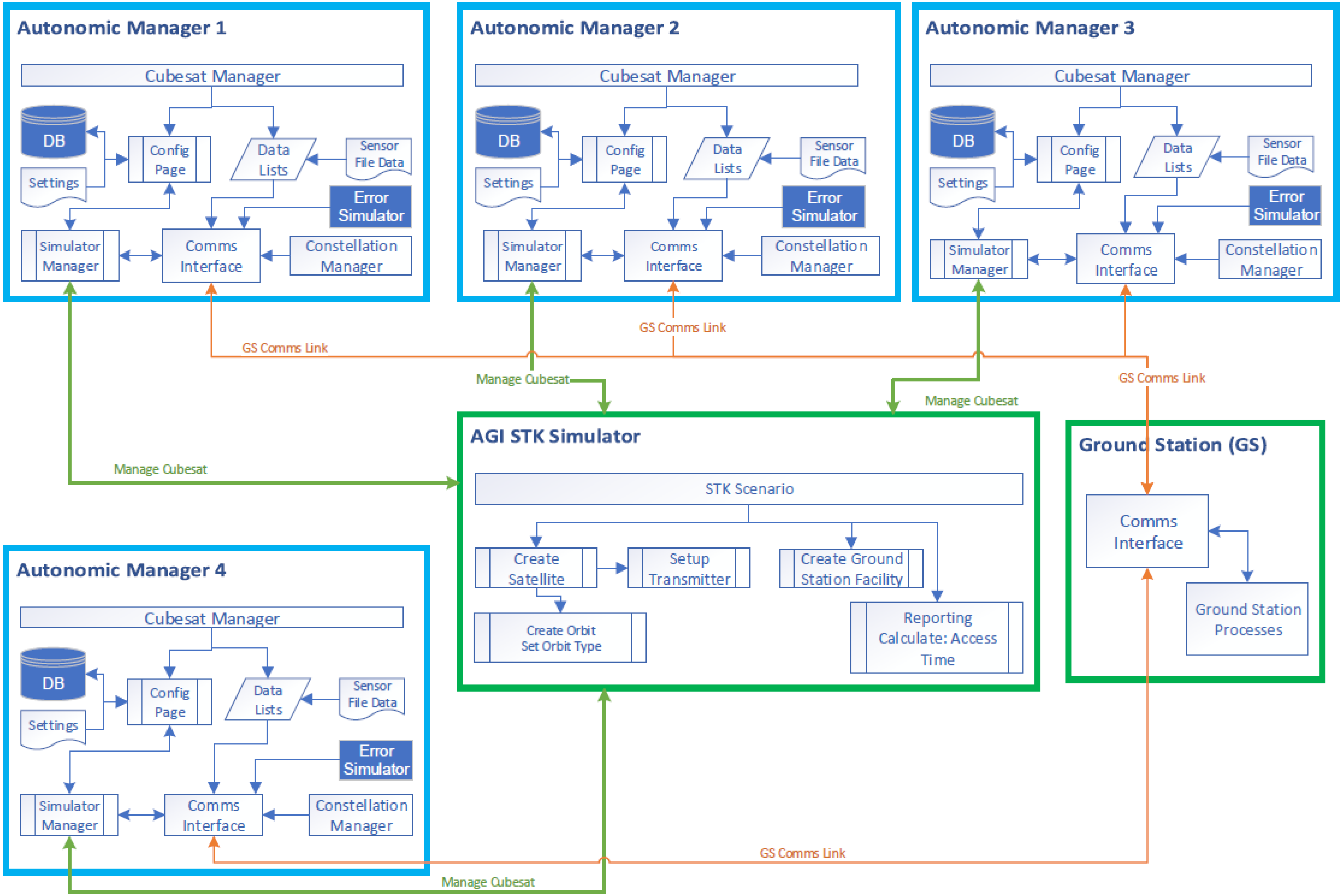


CubeSat Autonomic Capability Level 4 (CubeSat)

Cubesat Autonomic Capability Model Level 4

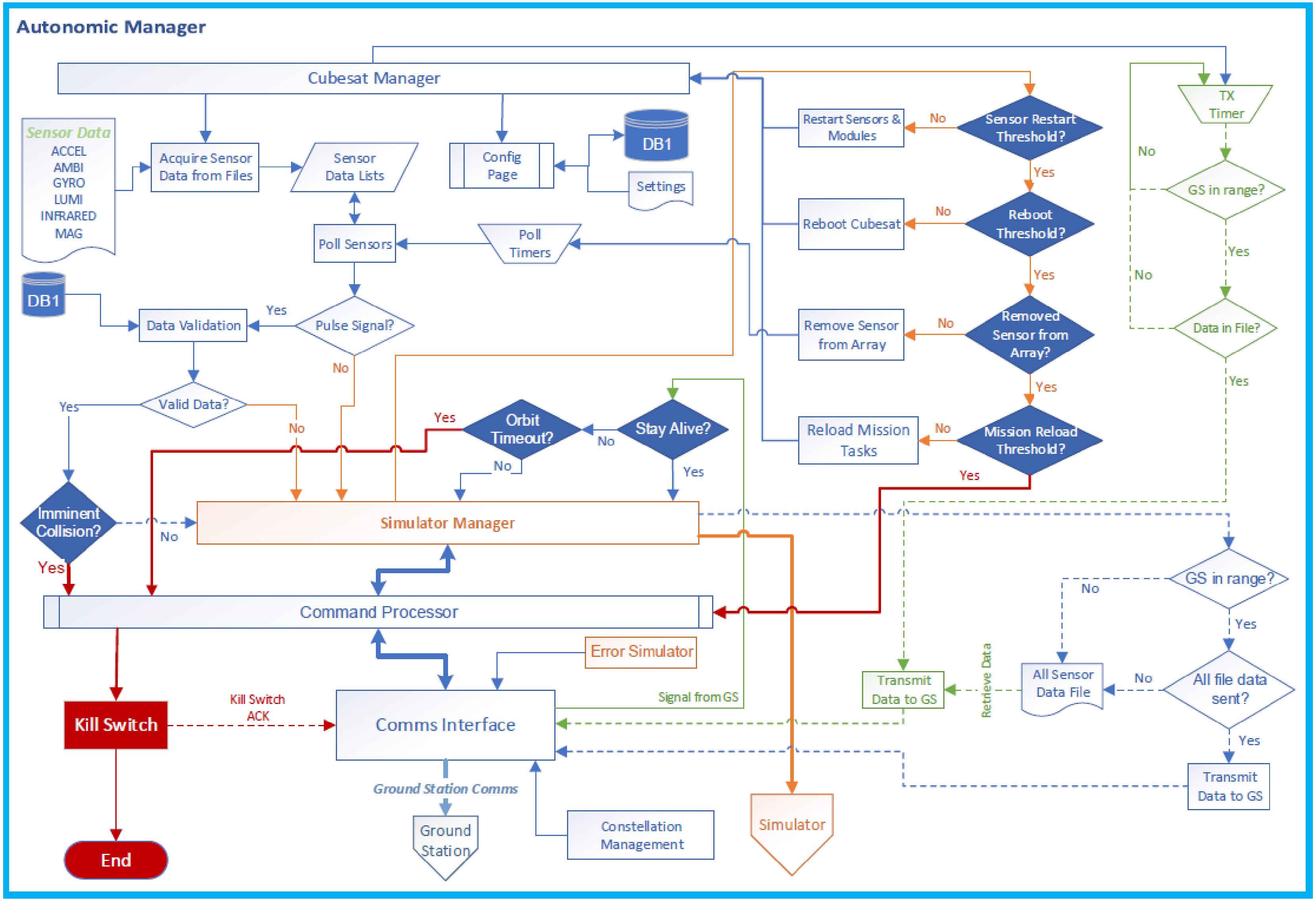


CubeSat Autonomic Capability Level 4 & 5 (Constellation CubeSats)

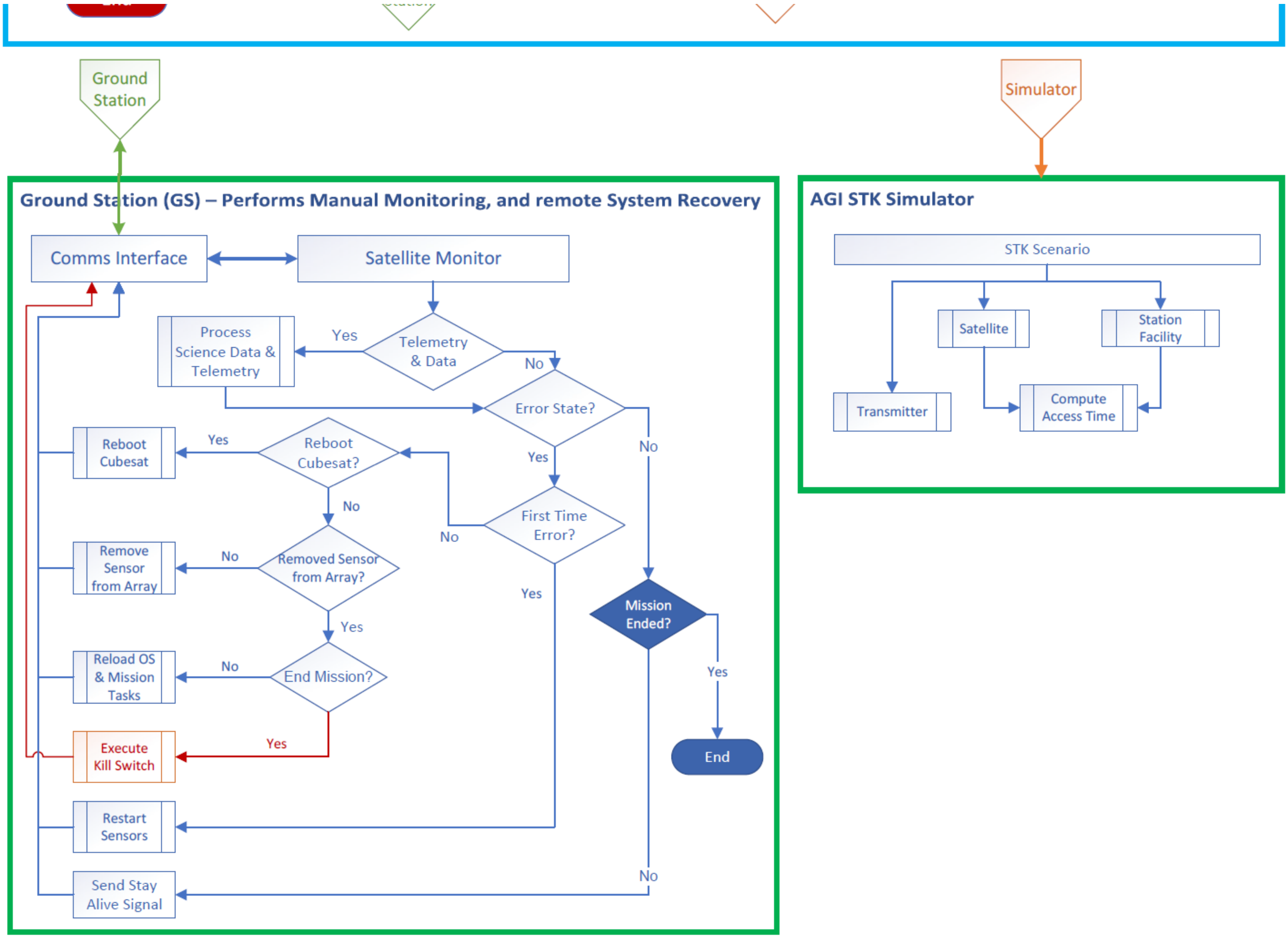


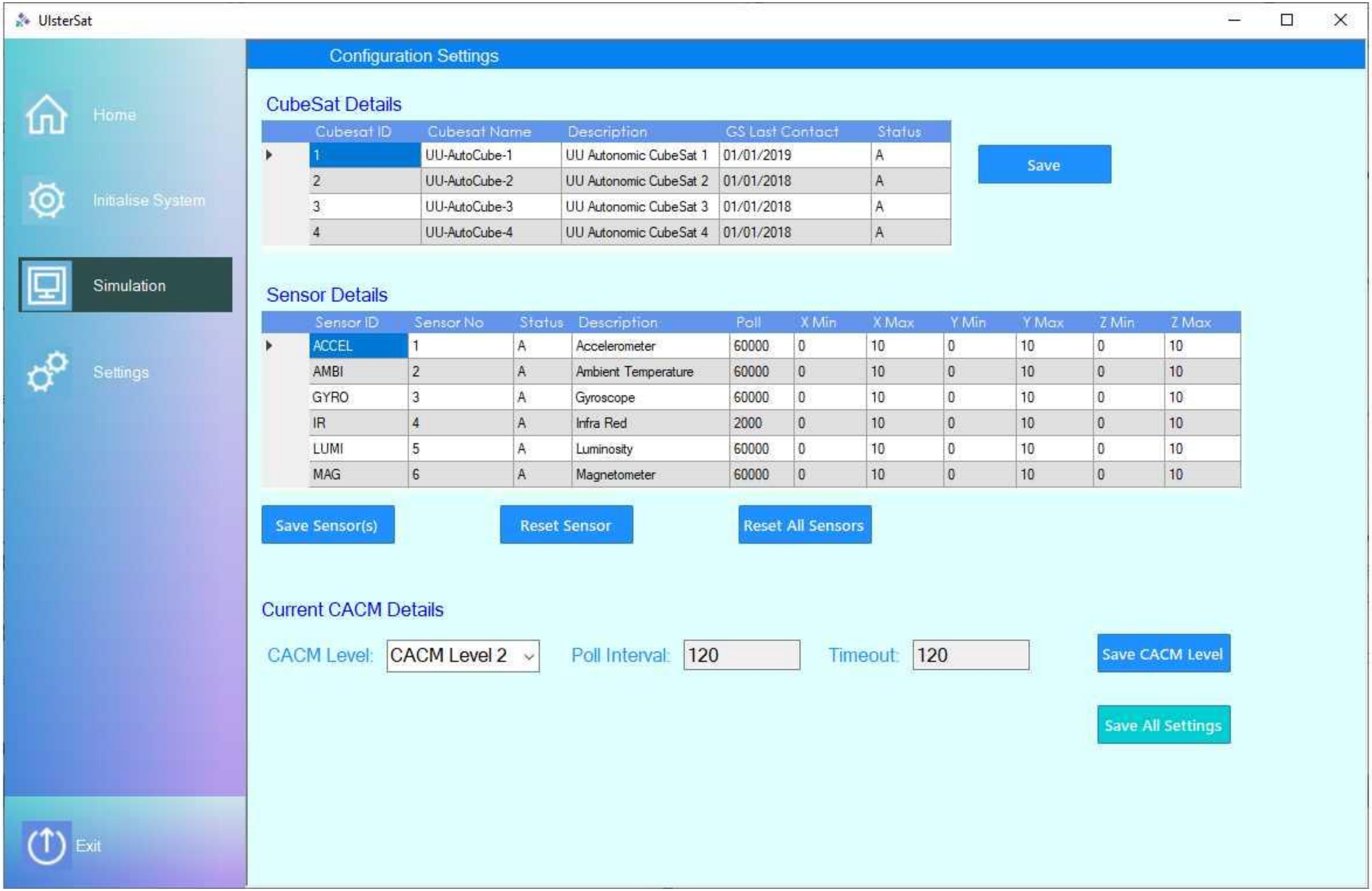
CubeSat Autonomic Capability Level 5 (CubeSat)

Cubesat Autonomic Capability Model Level 5



CubeSat Autonomic Capability Level 1 (Ground Station)



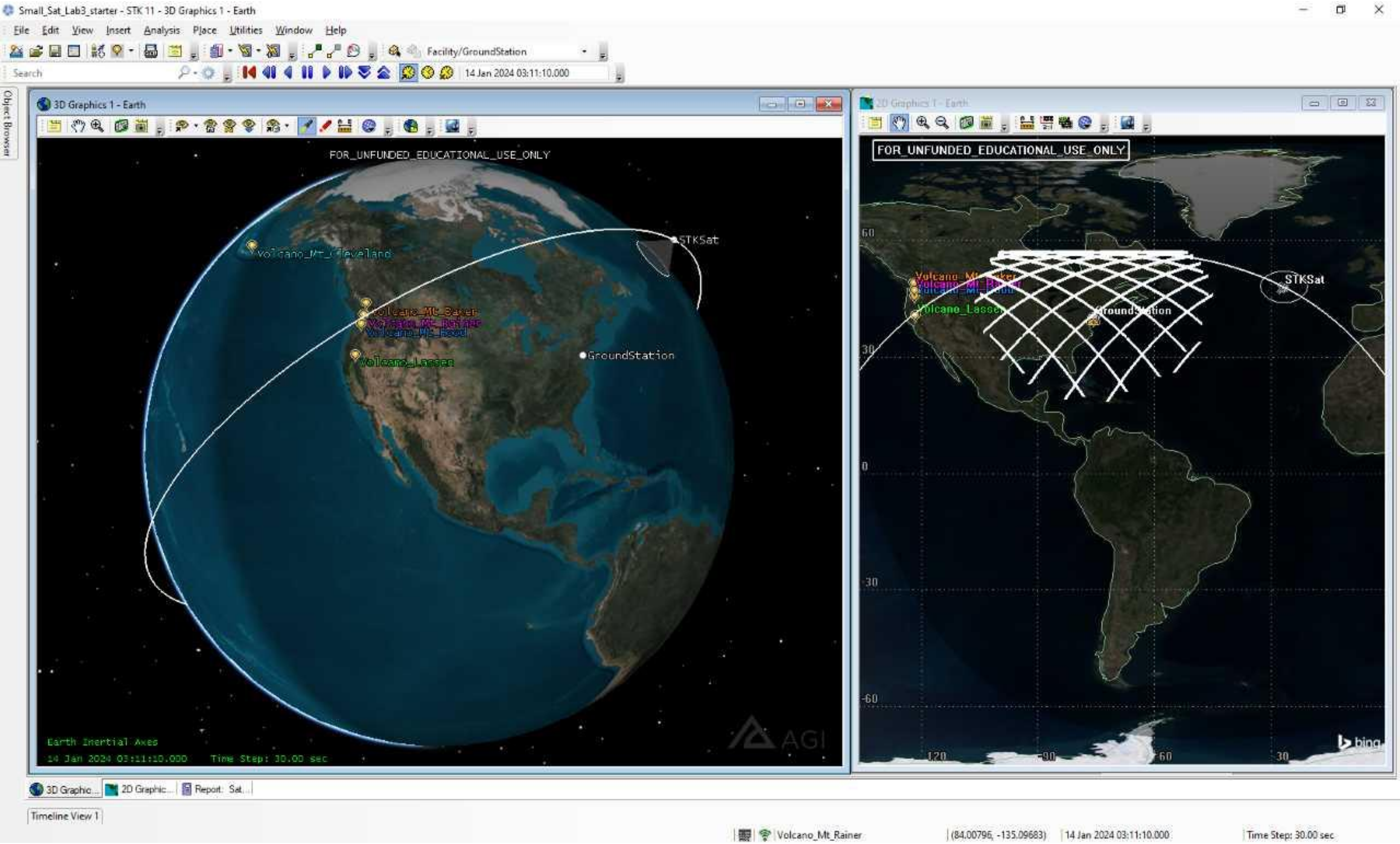

UlsterSat
Home
Initialise System
Simulation
Settings
Exit
Configuration Settings
CubeSat Details

Cubesat ID	Cubesat Name	Description	GS Last Contact	Status
1	UU-AutoCube-1	UU Autonomic CubeSat 1	01/01/2019	A
2	UU-AutoCube-2	UU Autonomic CubeSat 2	01/01/2018	A
3	UU-AutoCube-3	UU Autonomic CubeSat 3	01/01/2018	A
4	UU-AutoCube-4	UU Autonomic CubeSat 4	01/01/2018	A

Save
Sensor Details

Sensor ID	Sensor No	Status	Description	Poll	X Min	X Max	Y Min	Y Max	Z Min	Z Max
ACCEL	1	A	Accelerometer	60000	0	10	0	10	0	10
AMBI	2	A	Ambient Temperature	60000	0	10	0	10	0	10
GYRO	3	A	Gyroscope	60000	0	10	0	10	0	10
IR	4	A	Infra Red	2000	0	10	0	10	0	10
LUMI	5	A	Luminosity	60000	0	10	0	10	0	10
MAG	6	A	Magnetometer	60000	0	10	0	10	0	10

Save Sensor(s)
Reset Sensor
Reset All Sensors
Current CACM Details
CACM Level:
CACM Level 2
Poll Interval:
120
Timeout:
120
Save CACM Level
Save All Settings



Small_Sat_Lab3_starter - STK 11 - 3D Graphics 1 - Earth
File Edit View Insert Analysis Place Utilities Window Help
Facility/GroundStation
Search
14 Jan 2024 03:11:10.000
3D Graphics 1 - Earth
FOR_UNFUNDED_EDUCATIONAL_USE_ONLY
STKSat
Volcano_Mt_Cleveland
Volcano_Mt_Baker
Volcano_Mt_Rainer
Volcano_Mt_Hood
Volcano_Lassen
GroundStation
Earth Inertial Axes
14 Jan 2024 03:11:10.000 Time Step: 30.00 sec
AGI
2D Graphics 1 - Earth
FOR_UNFUNDED_EDUCATIONAL_USE_ONLY
STKSat
Volcano_Lassen
60
30
0
-30
-60
120
90
60
30
bing
3D Graphic...
2D Graphic...
Report: Sat...
Timeline View 1
Volcano_Mt_Rainer
(84.00796, -135.09683)
14 Jan 2024 03:11:10.000
Time Step: 30.00 sec
Object Browser

Conclusion, Summary & Discussion

The hypothesis presented in this talk/paper, was that **Trustworthy Autonomous Systems (TAS)** and **Assured Autonomous Systems** can be (*partially*) achieved through **Autonomic Computing extended with Apoptotic Computing**.

The research carried out in the noughties on Autonomic and Apoptotic Computing with NASA GSFC, briefly recapped here, started in the first instance as expanding on the NASA Formal Approaches to Swarm Technologies (FAST) project which was funded by the NASA Office of Systems and Mission Assurance (OSMA) through its Software Assurance Research Program (SARP). The concern that was attempting to address here was the future concept missions of potentially 1000s of autonomous adaptive craft and how can you assure their operation. The apoptotic (pre-programmed nano-craft death) became the ultimate assurance with autonomic paradigm ensuring the trustworthy self-management of the mission assets. This work lead to 16 patents [43], such as [44].

This assurance and trustworthy via autonomic and apoptotic computing theme carried on throughout reflection on our other research; from elderly care smart homes to prison systems, robotics and returning to space with cubesats and the derivation of a generic architecture and a capability model.

Yet the larger, more difficult task of combining these point solutions into wider autonomous systems remains. More consideration must be given to integrating solutions, and to choosing solutions from the range of possibilities— to trustworthy and assured autonomous and autonomic systems engineering, in other words.

Without the development of such an approach, we will simply rediscover the risks of feature interaction at a higher level, and in a way that is so dynamic as to be resistant to debugging and testing. We are confident, however, that the foundation exists to construct a systems theory and practice from which we can engineer trustworthy autonomous solutions for the next generation of enterprise and sensor systems.

We can only achieve so much…

Roy Sterritt
@RoysterUlster
r.sterritt@ulster.ac.uk

Thank you.
Questions?
Comments?

Acknowledgements

Some of the research described in this presentation is patented by Roy Sterritt (Ulster University) and Mike Hinchey (Lero—the Irish Software Research Centre) through NASA and assigned to the US government.

Thanks to all my colleagues and in particular my PhD students who have done a lot of the heavy lifting, in particular during the twenty-tens.